Iterative receiver for MIMO-OFDM systems based on sphere decoding : convergence, performance and complexity tradeoffs

Rida El Chall

Résumé

Pour permettre l'accroissement de débit et de robustesse dans les futurs systèmes de communication sans fil, les processus itératifs sont de plus considérés dans les récepteurs.
Cependant, l'adoption d'un traitement itératif pose des défis importants dans la conception du récepteur. Dans cette thèse, un récepteur itératif combinant les techniques de détection multi-antennes avec le décodage de canal est étudié. Trois aspects sont considérés dans un contexte MIMO-OFDM: la convergence, la performance et la complexité du récepteur.

Dans un premier temps, nous étudions les différents algorithmes de détection MIMO à décision dure et souple basés sur l'égalisation, le décodage sphérique, le décodage K-Best et l'annulation d'interférence. Un décodeur K-best de faible complexité (LC-K-Best) est proposé pour réduire la complexité sans dégradation significative des performances. Nous analysons ensuite la convergence de la combinaison de ces algorithmes de détection avec différentes techniques de codage de canal, notamment le décodeur turbo et le décodeur LDPC en utilisant le diagramme EXIT. En se basant sur cette analyse, un nouvel ordonnancement des itérations internes et externes nécessaires est proposé. Les performances du récepteur ainsi proposé sont évaluées dans différents modèles de canal LTE, et comparées avec différentes techniques de détection MIMO. Ensuite, la complexité des récepteurs itératifs avec différentes techniques de codage de canal est étudiée et comparée pour différents modulations et rendement de code. Les résultats de simulation montrent que les approches proposées offrent un bon compromis entre performance et complexité.

D'un point de vue implémentation, la représentation en virgule fixe est généralement utilisée afin de réduire les coûts en termes de surface, de consommation d'énergie et de temps d'exécution. Nous présentons ainsi une représentation en virgule fixe du récepteur itératif proposé basé sur le décodeur LC K-Best. En outre, nous étudions l'impact de l'estimation de canal sur la performance du système. Finalement, le récepteur MIMO-OFDM itératif est testé sur la plateforme matérielle WARP, validant le schéma proposé.

Mots-clés: récepteur itératif, MIMO, décodeur sphérique, décodeur K-Best, MMSE-IC, V-BLAST, décodeur Turbo, décodeur LDPC, virgule fixe, estimation de canal, synchronisation.

Abstract

Recently, iterative processing has been widely considered to achieve near-capacity performance and reliable high data rate transmission, for future wireless communication systems. However, such an iterative processing poses significant challenges for efficient receiver design. In this thesis, iterative receiver combining multiple-input multiple-output (MIMO) detection with channel decoding is investigated for high data rate transmission. The convergence, the performance and the computational complexity of the iterative receiver for MIMO-OFDM system are considered.

First, we review the most relevant hard-output and soft-output MIMO detection algorithms based on sphere decoding, K-Best decoding, and interference cancellation. Consequently, a low-complexity K-best (LC-K-Best) based decoder is proposed in order to substantially reduce the computational complexity without significant performance degradation. We then analyze the convergence behaviors of combining these detection algorithms with various forward error correction codes, namely LTE turbo decoder and LDPC decoder with the help of Extrinsic Information Transfer (EXIT) charts. Based on this analysis, a new scheduling order of the required inner and outer iterations is suggested. The performance of the proposed receiver is evaluated in various LTE channel environments, and compared with other MIMO detection schemes. Secondly, the computational complexity of the iterative receiver with different channel coding techniques is evaluated and compared for different modulation orders and coding rates. Simulation results show that our proposed approaches achieve near optimal performance but more importantly it can substantially reduce the computational complexity of the system.

From a practical point of view, fixed-point representation is usually used in order to reduce the hardware costs in terms of area, power consumption and execution time. Therefore, we present efficient fixed point arithmetic of the proposed iterative receiver based on LC-K-Best decoder. Additionally, the impact of the channel estimation on the system performance is studied. The proposed iterative receiver is tested in a real-time environment using the MIMO WARP platform.

Keywords: Iterative receiver, MIMO, Sphere decoder, K-Best decoder, MMSE-IC, V-BLAST, Turbo decoder, LDPC decoder, fixed-point arithmetic, channel estimation, time synchronization.

INSA RENNES

N° d'ordre : 15ISAR 23 / D15 - 23
Institut National des Sciences Appliquées de Rennes
20, Avenue des Buttes de Coëmes • CS 70839 • F-35708 Rennes Cedex 7
Tel : 02 23 23 82 00 - Fax : 02 23 23 83 96

Thèse

2015

Rida EL CHALL

THESE INSA Rennes
sous le sceau de l'Université européenne de Bretagne
pour obtenir le titre de
DOCTEUR DE L'INSA DE RENNES
Spécialité : Electronique et Télécommunications

présentée par
Rida El Chall
ECOLE DOCTORALE : *MATISSE*
LABORATOIRE : *IETR*

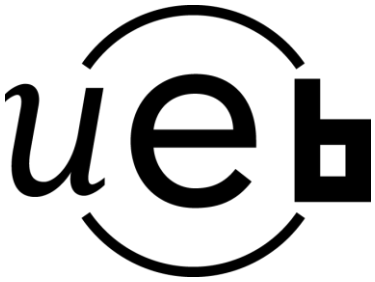
Récepteur itératif pour les systèmes MIMO-OFDM basé sur le décodage sphérique : Convergence, Performance et Complexité

Thèse soutenue le 22.10.2015
devant le jury composé de :

Guillaume Gelle
Professeur à l'Université de Reims Champagne-Ardenne / Président
Catherine Douillard
Professeur à Télécom Bretagne / Rapporteur
Christophe Jego
Professeur à l'INP Bordeaux (ENSEIRB-MATMECA) / Rapporteur
Olivier Berder
Professeur à l'IUT Lannion (UR1) / Examinateur
Maryline Hélard
Professeur à l'INSA de Rennes / Co-directrice de thèse
Fabienne Nouvel
Maître de conférences, HDR à l'INSA de Rennes / Directrice de thèse

Récepteur itératif pour les systèmes MIMO-OFDM basé sur le décodage sphérique : Convergence, Performance et Complexité

Rida El Chall



En partenariat avec

Dedication

To my beloved parents

Acknowledgments

The success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my sincere gratitude to all people who have been contributed in the successful of this thesis.

First and foremost, I would like to express my greatest appreciation to my supervisor, Fabienne Nouvel. I am extremely thankful to your always availability and your precious supports in my research and daily life. Thank you also for all the encouragement, patience and kindness. I would also to thank my co-supervisor, Maryline Hélard, for her invaluable guidance and generous advises throughout the last three years.

I am deeply grateful to my colleague, Ming Liu, for the useful suggestions, for the insightful advises and for the inspiring discussions.

It is also my pleasure to express my sincere thanks to the committee members, Prof. Christophe Jego, Prof. Catherine Douillard, Prof.Guillaume Gelle and Prof. Olivier Berder, for reading and evaluating my manuscript. Their involvement and expert insights have greatly helped me to enrich my thesis.

Besides, I would like to thank the staff members in IETR, for providing a friendly working environment and facilities to complete this thesis. I also wish to express my gratitude to the officials and other staff members who rendered their help during the period of my thesis work.

In addition to the above support, I received equally important assistance from family and friends. My gratitude goes to my beloved parents, my sister and my brother for their understanding, endless love and continuous encouragement. Last but not least to my friends for their kindness, moral support and all the fun we have during the past years.

Abstract

Recently, iterative processing has been widely considered to achieve near-capacity performance and reliable high data rate transmission, for future wireless communication systems. However, such an iterative processing poses significant challenges for efficient receiver design. In this thesis, iterative receiver combining multiple-input multiple-output (MIMO) detection with channel decoding is investigated for high data rate transmission. The convergence, the performance and the computational complexity of the iterative receiver for MIMO-OFDM system are considered.

First, we review the most relevant hard-output and soft-output MIMO detection algorithms based on sphere decoding, K-Best decoding, and interference cancellation. Consequently, a low-complexity K-best (LC-K-Best) based decoder is proposed in order to substantially reduce the computational complexity without significant performance degradation. We then analyze the convergence behaviors of combining these detection algorithms with various forward error correction codes, namely LTE turbo decoder and LDPC decoder with the help of Extrinsic Information Transfer (EXIT) charts. Based on this analysis, a new scheduling order of the required inner and outer iterations is suggested. The performance of the proposed receiver is evaluated in various LTE channel environments, and compared with other MIMO detection schemes. Secondly, the computational complexity of the iterative receiver with different channel coding techniques is evaluated and compared for different modulation orders and coding rates. Simulation results show that our proposed approaches achieve near optimal performance but more importantly it can substantially reduce the computational complexity of the system.

From a practical point of view, fixed-point representation is usually used in order to reduce the hardware costs in terms of area, power consumption and execution time. Therefore, we present efficient fixed-point arithmetic of the proposed iterative receiver based on LC-K-Best decoder. Additionally, the impact of the channel estimation on the system performance is studied. The proposed iterative receiver is tested in a real-time environment using the MIMO WARP platform.

Keywords: Iterative receiver, MIMO, Sphere decoder, K-Best decoder, MMSE-IC, V-BLAST, Turbo decoder, LDPC decoder, fixed-point arithmetic, channel estimation, time synchronization.

Résumé

Pour permettre l'accroissement de débit et de robustesse dans les futurs systèmes de communication sans fil, les processus itératifs sont de plus considérés dans les récepteurs. Cependant, l'adoption d'un traitement itératif pose des défis importants dans la conception du récepteur. Dans cette thèse, un récepteur itératif combinant les techniques de détection multi-antennes avec le décodage de canal est étudié. Trois aspects sont considérés dans un contexte MIMO-OFDM: la convergence, la performance et la complexité du récepteur.

Dans un premier temps, nous étudions les différents algorithmes de détection MIMO à décision dure et souple basés sur l'égalisation, le décodage sphérique, le décodage K-Best et l'annulation d'interférence. Un décodeur K-best de faible complexité (LC-K-Best) est proposé pour réduire la complexité sans dégradation significative des performances. Nous analysons ensuite la convergence de la combinaison de ces algorithmes de détection avec différentes techniques de codage de canal, notamment le décodeur turbo et le décodeur LDPC en utilisant le diagramme EXIT. En se basant sur cette analyse, un nouvel ordonnancement des itérations internes et externes nécessaires est proposé. Les performances du récepteur ainsi proposé sont évaluées dans différents modèles de canal LTE, et comparées avec différentes techniques de détection MIMO. Ensuite, la complexité des récepteurs itératifs avec différentes techniques de codage de canal est étudiée et comparée pour différents modulations et rendement de code. Les résultats de simulation montrent que les approches proposées offrent un bon compromis entre performance et complexité.

D'un point de vue implémentation, la représentation en virgule fixe est généralement utilisée afin de réduire les coûts en termes de surface, de consommation d'énergie et de temps d'exécution. Nous présentons ainsi une représentation en virgule fixe du récepteur itératif proposé basé sur le décodeur LC K-Best. En outre, nous étudions l'impact de l'estimation de canal sur la performance du système. Finalement, le récepteur MIMO-OFDM itératif est testé sur la plateforme matérielle WARP, validant le schéma proposé.

Mots-clés: récepteur itératif, MIMO, décodeur sphérique, décodeur K-Best , MMSE-IC, V-BLAST, décodeur Turbo, décodeur LDPC, virgule fixe, estimation de canal, synchronisation.

Contents

List of acronyms

The following list summarizes the acronyms used in this thesis.

3GPP	3^{rd} Generation Partnership Project
ADC	Analog-to-Digital Converter
AGC	Automatic Gain Control
APP	A *Posteriori* Probability
AWGN	Additive White Gaussian Noise
BCJR	Bahl, Cocke, Jelinek and Raviv
BER	Bit Error Rate
BICM	Bit Interleaved Coded Modulation
BLAST	Bell Laboratories Layered Space-Time
BP	Belief Propagation
BPSK	Binary Phase-Shift Keying
CAI	Co-Antenna Interference
CAZAC	Constant Amplitude Zero Auto-Correlation
CB	Circular Buffer
CDMA	Code Division Multiple Access
CE	Channel Estimation
CFO	Carrier Frequency Offset
CFR	Channel Frequency Response
CIR	Channel Impulse Response
CL	Closed Loop
CN	Check Node
cn	Condition number
CP	Cyclic Prefix
CORDIC	Coordinate rotation digital computer
CSI	Channel State Information
CVP	Closest Vector Point
DA	Data-Aided
DAC	Digital-to-Analog Converter
D-BLAST	Diagonal-Bell Laboratories Layered Space-Time
DCT	Discrete Cosine Transform
DDCE	Decision Directed Channel Estimation
DFE	Decision-Feedback Equalizer
DFT	Discrete Fourier Transform
DSP	digital signal processing
DVB-T	Digital Video Broadcasting- Terrestrial

EM	Expectation Maximization
EPA	Extended Pedestrian A
ETU	Extended Typical Urban
EVA	Extended Vehicular A
EXIT	EXtrinsic Information Transfer
FD	Frequency domain
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FP	Fincke-Pohst
FPGA	Field Programmable Gate Array
flops	FLoating-point Operations Per Second
FSD	Fixed Sphere Decoder
GI	Guard Interval
GR	Givens Rotations
GS	Gram-Schmidt
H-BLAST	Horizontal-Bell Labs Advanced Space-Time Transmission
HH	HouseHolder
IC	Interference Canceller
ICI	Inter-Carrier Interference
i.i.d	independent and identically distributed
IDFT	Inverse Discrete Cosine Transform
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
ISM	Industrial, scientific and medicals
ISI	Inter-Symbol Interference
LA	LLL Algorithm
LC-K-Best	Low-complexity K-Best
LD	Linear Detector
LDC	Linear Dispersion Code
LDPC	Low Density Parity Check (code)
LLR	Log-Likelihood Ratio
LMMSE	Linear Minimum Mean Square Error
LR	Lattice Reduction
LS	Least Squares
LSD	List Sphere Decoder
LTE-(A)	Long-Term Evolution-(Advanced)
LU	Lower Upper
LUT	Look-Up Table
MAP	Maximum A *Posteriori*
MCS	Modulation and Coding Schemes
MIMO	Multiple-input Multiple-output
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MMSE-IC	Minimum Mean Square Error Interference Canceller
MRC	Maximum Ratio Combining
MSA	Minimum Sum Algorithm
MSE	Mean Square Error

OFDM	Orthogonal Frequency Division Multiplexing
OL	Open Loop
OSIC	Ordered Successive Interference Cancellation
OSTBC	Orthogonal Space Time Block Code
PAPR	Peak-to-Average Power Ratio
PED	Partial Euclidean distance
PIC	Parallel Interference cancellation
PN	Pseudo Noise
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QPP	Quadratic permutation polynomial
QRD	QR Decomposition
RLS	Recursive least squares
RF	Radio Frequency
RM	Rate Matching
RS	Reed-Solomon
RTS	Repeated Tree Search
SAGE	space-alternating generalized expectation-maximization
SA	Seysen's Algorithm
SD	Sphere Decoder
SE	Schnorr-Euchner
SIC	Successive Interference Cancellation
SISO	Soft-Input Soft-Output
SM	Spatial Multiplexing
SNR	Signal-to-Noise Ratio
SOVA	Soft Output Viterbi Algorithm
SPA	Sum Product Algorithm
SQRD	Sorted QR Decomposition
STBC	Space Time Block Code
ST-BICM	Space Time Bit-Interleaved Coded Modulation
STTC	Space Time Trellis Code
SVD	Singular Value Decomposition
TSVD	Truncated Singular Value Decomposition
V-BLAST	Vertical-Bell Laboratories Layered Space-Time
VLSI	Very Large Scale Integration
VN	Variable Node
WARP	Wireless Open-Access Research Platform
WIMAX	Wireless Interoperability for Microwave Access
WLAN	Wireless Local Access Network
ZF	Zero-Forcing

Notations

Though out this thesis, matrices are set in boldface capital letters, vectors in boldface lowercase letters, scalars in lowercase letters. The mathematical notations and variables used in our work are summarized in the following table.

Mathematical Notations

$\mathrm{Re}\,(.)$	Real part of a complex variable
$\mathrm{Im}\,(.)$	Imaginary part of a complex variable
$\mathbf{A}^T$	Transpose of matrix $\mathbf{A}$
$\mathbf{A}^H$	Conjugate transpose of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$
$\mathbf{A}^\dagger$	Pseudo-inverse of matrix $\mathbf{A}$
$A_{i,j}$	i^{th} row and j^{th} column entry of matrix $\mathbf{A}$
$A_{i,:}$	i^{th} row of matrix $\mathbf{A}$
$A_{:,j}$	j^{th} column of matrix $\mathbf{A}$
$\mathrm{Vec}(\mathbf{A})$	Vectorization of matrix $\mathbf{A}$
$\mathrm{tr}(\mathbf{A})$	Trace of matrix $\mathbf{A}$
$\mathrm{rank}(\mathbf{A})$, $r_{\mathbf{A}}$	Rank of matrix $\mathbf{A}$
$\mathbf{I}_n$	Identity matrix of size n
$\mathbf{0}_{m\times n}$	Zero matrix of dimension $m \times n$
$\mathbf{e}_k$	Unitary vector (zeros elements except the k^{th} element is one)
$i = \sqrt{-1}$	Imaginary part
$\exp(x)$, e^x	Exponential function
$\log$	Naperian logarithm function base e
$\log_2$	Logarithm function base 2
$\log_{10}$	Logarithm function base 10
$\delta(\hat{A}\mathring{u})$	Dirac delta function
$\mathbb{E}\{x\}$	Expectation of a random variable x
$P(x)$	Probability of a random variable x
$p(x)$	Probability density function of x
$\|x\|^2$	l^2-norm (Euclidean distance) of vector x
$\|x\|$	Absolute value of the variable x
$\sqrt{(.)}$	Square root
$I(x,y)$	Mutual information between x and y
σ_x^2	Variance of the a random variable x
$\lceil a \rceil$	Nearest integer equal or greater than a
$\lfloor a \rfloor$	Nearest integer equal or less than a

$\mathcal{Q}\{.\}$	Quantization to the nearest constellation symbol
$\mathcal{N}(\mu, \sigma^2)$	Normal law of mean μ and σ^2
$J_0(.)$	Zero order Bessel function
$\otimes$	Kronecker product
$\max(x, y)$	Maximum of the arguments
$\min(x, y)$	Minimum of the arguments

Variables

P_L	Pass loss
T_s	Signal period
B_s	Signal bandwidth
T_c	Coherence time
B_c	Coherence bandwidth
f_c	Carrier frequency
f_d	Doppler frequency
f_m	Maximum Doppler frequency
v	User velocity
c	Speed of light
L	Number of channel taps
$h_{l,k}$	l^{th} tap of the channel impulse response at k^{th} instant
α_l	Attenuation of the l^{th} tap
τ_l	Delay of the l^{th} tap
$H(f, k)$	Channel frequency response
τ_{rms}	Root mean square delay of the channel
$\tau_{\max}$	Maximum delay spread of the channel
T_e	Sampling period
C	Channel capacity
M_c	Constellation order, $M_c = 2^Q$
Q	Number of bit per symbol,$Q = \log_2(M_c)$
$\chi_{i,b}$	Sets of symbol vectors corresponding to the i^{th} symbol and b^{th} bit
R_c	Coding rate K_b/N_b
K_b	Information block length
N_b	Codeword length
M_b	Parity bits $N_b - K_b$
γ	Branch metric
α	Forward recursive metric
β	Backward recursive metric
m	Memory length of the component encoder
L_e	Extrinsic LLR
L_a	*a priori* LLR
L_{sys}	Systematic LRR
$\mathbf{H}_{\text{p}}$	Parity check matrix of LDPC
L_v	Variable node message
L_c	Check node message
I_{in}	Number of inner iterations
I_{out}	Number of outer iterations

N_t	Number of transmit antennas
N_r	Number of receive antennas
R_s	STBC rate
Q_s	Number of input to STBC
T	Time of STBC
G_{SM}	SM gain
G_{d}	Diversity gain
$\mathbf{F}$	DFT matrix
N	FFT size
N_c	Number of useful (modulated) sub-carriers
Δ	Guard Interval length
u_k	Sequence of information bits
c_k	Sequence of coded bits
$\mathbf{y}$	Received signal vector
$\tilde{\mathbf{y}}$	Modified received signal vector
$\mathbf{n}$	noise signal vector
$\mathbf{s}$	transmitted signal vector
$\mathbf{H}$	Channel matrix
$\mathbf{R_H}$	Correlation channel matrix
r_s	Sphere radius
d_1	Euclidean distance metric
d_i	Partial Euclidean distance metric
e_i	Branch metric
m^A	*a priori* based metric
m^C	Channel based metric
K	Number of candidates in K-Best
L_{Clip}	LLR clipping level
$\mathbf{Q}$	Unitary matrix resulting from QRD
$\mathbf{R}$	Triangular matrix resulting from QRD
$\mathbf{T}$	Unimodular matrix
$\mathbf{P}$	Permutation matrix
$\tilde{s}_i$	Equalized or detected symbol
$\hat{s}_i$	Estimated symbol
β_i	Equalizer bias
η_i	Residual noise plus interference
N_p	Number of pilot sub-carrier
Δf_p	Pilot spacing in the frequency domain
Δt_p	Pilot spacing in the time domain
$\mathbf{R}_{H_pH_p}$	Auto-correlation channel matrix at pilot position
$\mathbf{R}_{HH_p}$	Cross-correlation matrix of $\mathbf{H}$ and $\mathbf{H}_p$
$\hat{\mathbf{H}}$	Estimated channel matrix
$\mathbf{E}$	Channel estimation error

List of Figures

List of Tables

Résumé étendu en français

Introduction

Au cours des dernières décennies, les systèmes de communications ont connu une grande évolution et sont devenus vitaux dans notre vie quotidienne. Cette grande évolution est liée à la demande des utilisateurs pour une transmission fiable avec un grand débit et une bonne qualité de service. Cependant, des nombreux défis se posent pour augmenter le débit de transmission tels que la puissance de transmission, la bande de fréquence limitée et les évanouissements de canal.

Pour cela, plusieurs techniques avancées ont été récemment développées pour relever ces défis. Parmi les techniques, on trouve les techniques multi-antennes, les techniques de codage du canal (turbo codes, LDPC codes) et les techniques multi-porteuses. Ces techniques ont été adoptées dans les standards comme 3GPP-LTE(A) pour les réseaux mobiles, IEEE 802.11n/ac et 802.16e/m pour les réseaux sans fil locaux et étendus, DVB-RCS et DVB-T2 pour la diffusion de vidéo numérique.

Les technologies multi-antennes (MIMO) consistent à transmettre et recevoir les données sur plusieurs antennes offrant un gain en capacité grâce à une exploitation de la diversité spatiale du canal de transmission. En outre, les techniques multi-porteuses (OFDM) permettent de lutter contre la sélectivité fréquentielle du canal et assurent une meilleure efficacité spectrale. Alors que, les techniques de codage de canaux modernes, tels que les turbo codes ou les codes LDPC, sont capables de protéger l'intégrité des données transmises et d'approcher la capacité du canal.

Par conséquent, la combinaison des techniques multi-antennes et multi-porteuses avec les techniques de codage de canal est devenue une solution attractive pour augmenter le débit et la robustesse des futurs systèmes de communications. Cependant, ces techniques complexifient le processus de réception. Il est désormais nécessaire de développer un récepteur avancé pour pouvoir récupérer les symboles transmis et gérer les interférences.

Ainsi pour un système MIMO-OFDM codé, la solution de réception optimale est de détecter et décoder conjointement les symboles reçus. Mais cette solution est inenvisageable pour les systèmes réels. Le processus itératif est alors adopté pour se rapprocher de la performance optimale en effectuant la détection et le décodage de canal séparément, tout en échangeant itérativement des informations souples entre eux. Malgré les performances intéressantes du récepteur itératif, il présente une complexité élevée qui dépend des algorithmes de détection MIMO, des techniques de décodage de canal et du nombre d'itérations effectuées. Afin de réduire la complexité, il faut néanmoins considérer des techniques de détection sous-optimales et optimiser le nombre d'itérations entre le détecteur et le décodeur pour un bon compromis entre performance et complexité.

La technique de détection optimale est basée sur l'algorithme de maximum de vraisem-

blance. Cette technique présente une complexité exponentielle en fonction de nombre d'antennes et de l'ordre de modulation. Pour cette raison, plusieurs algorithmes de détection sous-optimaux sont proposés dans la littérature allant de techniques à décisions dures vers des techniques à décisions souples.

Les travaux réalisés dans le cadre de cette thèse consistent à étudier les aspects algorithmiques ainsi que les aspects d'implémentation du récepteur itératif basé sur le décodage sphérique pour les systèmes MIMO-OFDM. Ainsi, une étude de convergence, de performance et de complexité des différentes techniques de détection et de décodage de canal est effectuée.

Dans un premier temps, nous étudions les différentes techniques de décodage de canal de détection basées sur l'égalisation et le décodage sphérique à décisions dures et à décisions douces. Ainsi, un nouveau décodeur K-Best (LC-K-Best) est proposé pour réduire la complexité du décodeur sphérique. Puis, une analyse de convergence de traitement itératif en utilisant le diagramme EXIT est réalisée pour les différentes techniques de détection et de décodage de canal afin d'optimiser le nombre d'itérations. Des simulations se basant sur les paramètres des applications (LTE notamment) sont lancées pour tester et comparer les performances de ces différentes techniques.

Dans un deuxième temps, la complexité des différents algorithmes de détection et de décodage de canal est étudiée en termes de nombre d'opérations. Une comparaison de complexité du récepteur itératif avec différentes techniques de détection et de décodage de canal est ainsi effectuée. En outre, une représentation du récepteur proposé en virgule fixe est présentée.

De plus, l'estimation de canal constitue un autre défi pour les systèmes MIMO-OFDM. Le récepteur doit estimer le canal pour chaque sous-porteuse et pour chaque antenne. Pour cette raison, plusieurs techniques d'estimation de canal sont étudiées et comparées. En outre, la synchronisation du système MIMO-OFDM est une autre clé qui doit être résolue dans le cas d'une transmission réelle pour pouvoir détecter le début de la trame. Une présentation assez brève de la synchronisation est ainsi décrite. Enfin, un test sur une plate-forme matérielle est réalisé pour prouver les performances du récepteur proposé.

Dans la suite, nous résumons en français les éléments importants détaillés dans les chapitres du manuscrit de thèse.

Chapitre 1 : Préliminaires

Ce chapitre donne un aperçu global des concepts basiques utilisés dans les chapitres suivants. Tout d'abord, la chaine de transmission numérique ainsi que les caractéristiques du canal de propagation radioélectrique sont présentés.

En général, toute chaine de transmission numérique peut être décomposée en trois blocs, à savoir, l'émetteur, le milieu de transmission et le récepteur. Le milieu de transmission constitue une contrainte pour le concepteur de systèmes, qui doit bien caractériser les paramètres du canal afin de prendre en compte les différentes fluctuations que peut subir un signal transmis. Ces fluctuations se classent en deux catégories : fluctuations à petites échelles et à grandes échelles. Plusieurs modèles de canal sont développés pour tenir compte de ces variations fréquentielles et temporelles [1]. Parmi les canaux théoriques, on cite le canal à bruit blanc et le canal de Rayleigh. Le canal à trajet multiples constitue un modèle de canal plus réel.

Après avoir introduit le modèle du canal, les techniques de codage de canal, notam-

ment le codage turbo [2] et le codage LDPC [3], sont décrites. Ces techniques modernes sont capables d'approcher les performances limites. Le turbo décodage peut être réalisé itérativement en utilisant l'algorithme de maximum à vraisemblance (MAP) [4] ou d'autres algorithmes sous-optimaux (Log-MAP, max-Log-MAP) [5]. Cet algorithme consiste à calculer les métriques de branches (γ), les métriques récursives d'aller et de retour (α et β). De même, le décodage LDPC est réalisé par l'algorithme de propagation de croyance qui échange itérativement des messages entres les nœuds de variables et les nœuds de parité [6].

Par la suite, nous présentons les techniques multi-porteuses et plus particulièrement la technique OFDM [7]. Cette technique est robuste vis-à-vis de la sélectivité fréquentielle des canaux de propagation et assure une meilleure efficacité spectrale. En effet, le canal sélectif en fréquence est décomposé en des sous-canaux orthogonaux non sélectifs en fréquence. Par ailleurs, un préfix cyclique de taille supérieure à l'étalement maximum des trajets est inséré pour annuler complètement l'interférence entre symboles. En plus, la technique OFDM est relativement facile à mettre en œuvre en utilisant la transformée de Fourier discrète et inverse (FFT et IFFT).

Une attention particulière est ensuite portée sur les techniques de transmission MIMO, le gain et la capacité de tels systèmes. En utilisant plusieurs antennes à l'émission et à la réception, un gain de diversité et un gain de multiplexage peuvent être obtenus en utilisant soit les techniques de codage espace-temps, soit les techniques de multiplexage spatial. D'autres techniques de précodage qui assurent un compromis entre les gains peuvent être aussi exploitées [8].

Enfin, nous décrivons notre modèle du système MIMO-OFDM constitué par N_t antennes d'emission et N_r antennes de réception, et nous présentons la problématique posée qui constitue la motivation de notre travail.

Chapitre 2 : Détecteurs MIMO à décision dure

Les systèmes MIMO sont considérés comme une solution attractive pour augmenter le débit et la robustesse des futurs systèmes de communication sans fil. Cependant, la plupart des techniques MIMO introduisent en réception de l'interférence co-antenne qui doit être annulée pour profiter au mieux de la diversité du canal MIMO. Dans ce chapitre, les différents algorithmes de détection MIMO à décision dure proposées dans la littérature sont rappelés. Ces techniques sont généralement classées en trois catégories : détecteurs linéaires, détecteurs à annulation d'interférence et décodeurs sphériques. Les avantages, les inconvénients et les performances de ces techniques sont ainsi présentés et comparés.

Détecteurs à maximum de vraisemblance

Le détecteur à maximum de vraisemblance (ML) constitue la solution optimale qui cherche à déterminer le vecteur transmis ($\mathbf{s}$) le plus probable [9]. Ce vecteur est celui qui minimise la distance euclidienne entre le vecteur reçu ($\mathbf{y}$) et toutes les combinaisons de symboles possibles :

$$\hat{\mathbf{s}}_{\mathrm{ML}} = \arg \min_{\mathbf{s} \in 2^{QN_t}} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2. \tag{1}$$

Malgré les performances optimales, ce détecteur présente une complexité qui croit exponentiellement avec le nombre d'antennes et l'ordre de modulation. C'est pourquoi, plusieurs détecteurs sous-optimaux sont étudiés dans la littérature.

Détecteurs linéaires

Le détecteur linéaire consiste simplement à appliquer un filtre linéaire sur le signal reçu [10]. Deux types de filtrage sont utilisés : le filtrage basé sur le forçage à zéro (ZF) et le filtrage basé sur la minimisation de l'erreur quadratique moyenne (MMSE).

L'égalisation de type ZF consiste à appliquer au signal reçu un filtre qui inverse l'effet du canal résultant en une annulation complète des interférences entre les symboles [11] :

$$\mathbf{G}_{\mathrm{ZF}} = \mathbf{H}^{\dagger} = (\mathbf{H}^{H}\mathbf{H})^{-1}\mathbf{H}^{H}. \tag{2}$$

Cependant, cette technique peut entrainer une augmentation du niveau de bruit et une dégradation des performances. Le filtrage MMSE est alors proposé. Il consiste à appliquer au signal reçu un filtre qui minimise l'erreur quadratique moyenne sur les vecteurs égalisés [12] :

$$\mathbf{G}_{\mathrm{MMSE}} = \left(\mathbf{H}^{H}\mathbf{H} + \frac{\sigma_n^2}{\sigma_s^2}\mathbf{I}_{N_t}\right)^{-1}\mathbf{H}^{H}, \tag{3}$$

où σ_n^2 et σ_s^2 représentent respectivement les puissances du bruit et du signal transmis.

Détecteurs à annulation d'interférence

Les détecteurs à annulation d'interférence, connus aussi sous le nom de détecteurs à retour de décision, utilisent les symboles préalablement détectés pour supprimer successivement les interférences et améliorer la détection des symboles à venir [11]. Deux variantes de ces détecteurs sont proposées : le détecteur à annulation d'interférence successive (SIC) connu aussi sous le nom de détecteur V-BLAST [13] et le détecteur basé sur la décomposition QR [14]. Ces types de détecteurs sont sensibles aux erreurs de propagation. En effet, une erreur effectuée lors de l'estimation d'un symbole entrainera par la suite des erreurs sur les symboles estimés. Pour éviter ces erreurs, il est préférable d'ordonner les symboles avant d'effectuer la détection, on parle alors de OSIC [13] ou de SQRD [14, 15].

Décodeurs sphériques

En utilisant la décomposition QR, le problème de détection se transforme en une recherche dans un treillis (arbre) [16]. On distingue deux familles de décodage par sphère : les algorithmes de type *depth-first search* comme le décodeur sphérique [17] ou *breadth-first search* comme le décodeur K-Best [18] ou FSD [19].

Depth-first search : décodeur sphérique (SD)

Le décodeur sphérique réduit la complexité du détecteur ML en limitant le nombre de vecteurs testés. Ainsi à chaque nœud de l'arbre, on vérifie que le vecteur testé est toujours contenu dans le rayon de la sphère (r_{s}). Si oui, les branches associées sont ainsi étudiées. Le traitement total d'une branche de treillis est effectué avant de traiter les autres :

$$\hat{\mathbf{s}}_{\mathrm{SD}} = \arg \min_{\mathbf{s}\in 2^{QN_t}} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \leq r_{\mathrm{s}}^2 \right\}. \tag{4}$$

M. Pohst propose une stratégie d'énumération (FP) des candidats sans aucun ordre pour éliminer les candidats à chaque étage du treillis [17]. Schnorr et Euchner (SE) proposent un raffinement de l'algorithme FP en appliquant un ordre de traitement des

candidats au niveau de chaque étage du treillis selon la distance par rapport à un point de référence [20].

Le paramètre principal du décodage par sphère est le rayon de la sphère. Si le rayon est grand, les performances sont améliorées mais le nombre de candidats testés est important. A l'inverse, si le rayon est petit, moins il y aura de candidats testés engendrant une dégradation des performances. Par ailleurs, le nombre de candidats traités n'est pas constant et dépend du signal reçu et du rapport signal à bruit ainsi que de la technique d'énumération adoptée.

Breadth-first search : décodeur K-Best

Le décodeur K-Best effectue une recherche de type breadth-first en traitant un nombre limité de meilleurs candidats (K) à chaque étage du treillis puis considère l'étage suivant [18]. Ainsi le nombre de candidats visités est constant au cours du temps offrant une complexité de décodage fixe. Les performances de ce type de détecteur sont cependant moins bonnes et dépendent du nombre de candidats traités.

Performances

Le tableau 4 présente la performance des différents détecteurs MIMO sans codage de canal. Le détecteur ML permet d'obtenir les meilleures performances, on trouve ensuite par ordre décroissant de performances les récepteurs SD, K-Best, FSD, OSIC, SIC, MMSE et ZF. Ainsi, plus le récepteur est complexe, meilleures sont les performances.

Table 4: Performance des detecteurs MIMO à decision dure.

Détecteur	Ordre de diversité	Perte de performance
ML	N_r	Optimale
ZF	$N_r - N_t + 1$	Très élevé
MMSE	$\approx N_r - N_t + 1$	élevé
SIC	$\approx N_r - N_t + 1$	faible
OSIC	$> N_r - N_t + 1 < N_r$	faible
SD	$\approx N_r$	$\approx$ Optimale
K-Best ($K \nearrow$)	$\approx N_r$	Proche de l'optimale
FSD	$\approx N_r$	Proche de l'optimale

Chapitre 3 : Détecteurs MIMO à décision souple

Ce chapitre est consacré à l'étude de la réception itérative pour les systèmes MIMO-OFDM codés. Après avoir présenté le principe de traitement itératif, nous présentons les techniques de détection MIMO à décision souple développées dans la littérature notamment la turbo-égalisation, le décodage sphérique par liste, le décodage single tree search et le décodage K-Best. Un décodeur K-Best avec une complexité réduite (LC-K-Best) est ainsi proposé. Ensuite, la convergence du récepteur itératif avec plusieurs techniques de détection et de décodage de canal (turbo, LDPC) est analysée afin de trouver le nombre d'itérations internes et externes nécessaires. Les performances de ces récepteurs sont évaluées dans un canal de Rayleigh et dans des canaux réels (LTE).

Principe turbo

Le principe turbo est initialement introduit par C. Berrou et A. Glavieux en 1993 pour les turbo-codes [2]. Ce principe est rapidement étendu sur l'ensemble des commu-

nications numériques. Il consiste à échanger des informations souples dites extrinsèques entre plusieurs fonctions (décodage, modulation, détection) afin d'améliorer les performances du système [21, 22].
Pour un système MIMO-OFDM codé, le détecteur MIMO et le décodeur de canal peuvent échanger des informations souples sur les bits codés. De plus, le décodeur de canal peut aussi échanger des informations sur les bits d'information dans le cas d'un turbo décodeur ou d'un décodeur LDPC. On désigne par I_{out} le nombre d'itérations externes entre le détecteur et le décodeur et par I_{in} le nombre d'itérations internes entre les composants de décodeur de canal.

Détecteur à maximum *a posteriori* (MAP)

Le détecteur à maximum *a posteriori* cherche à déterminer pour chaque bit le rapport logarithmique de vraisemblance (LLR) suivant :

$$L\left(x_{i,b}\right) = \log \frac{P(x_{i,b}=+1|\mathbf{y})}{P(x_{i,b}=-1|\mathbf{y})} = \log \frac{\sum_{s\in\chi_{i,b}^{+1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}{\sum_{\mathbf{s}\in\chi_{i,b}^{-1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}, \tag{5}$$

avec

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{\left(\pi N_0\right)^{N_r}} \exp\left(-\frac{1}{N_0}\left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2\right), \tag{6}$$

$$P(\mathbf{s}) = \prod_{i=1}^{N_t} P\left(s_i\right) = \prod_{i=1}^{N_t}\prod_{b=1}^{Q} P\left(x_{i,b}\right). \tag{7}$$

Afin de simplifier le calcul des LLRs, l'approximation max-log-MAP est généralement utilisée :

$$L\left(x_{i,b}\right) \approx \frac{1}{N_0}\min_{\chi_{i,b}^{-1}}\left\{d_1\right\} - \frac{1}{N_0}\min_{\chi_{i,b}^{+1}}\left\{d_1\right\}, \tag{8}$$

$$\begin{aligned} d_1 &= \left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2 - N_0 \log P(\mathbf{s}) \\ &= \left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2 - N_0 \sum_{i=1}^{N_t}\sum_{b=1}^{Q} \log P(x_{i,b}). \end{aligned} \tag{9}$$

L'inconvénient de ce détecteur est sa complexité exponentielle. En ce qui suit, plusieurs décodeurs à décision souple sont décrits pour réduire cette complexité.

Détecteurs souples à annulation d'interférence

Les détecteurs MIMO à annulation d'interférence sont basés sur le filtrage linéaire. Ils consistent à annuler les termes d'interférence entre les symboles en utilisant les symboles estimés. Cette suppression d'interférence peut être effectuée successivement comme en VBLAST [23] ou simultanément (parallèlement) comme dans le cas de l'égaliseur MMSE-IC [24, 25].

Egaliseur MMSE-IC

L'égaliseur MMSE-IC consiste à appliquer un filtre $\mathbf{p}_i$ sur le vecteur reçu et un filtre $\mathbf{q}_i$ sur le vecteur estimé. Le symbole égalisé $\tilde{s}_i$ s'exprime ainsi de la manière suivante [24] :

$$\tilde{s}_i = \mathbf{p}_i^H \mathbf{y} - \mathbf{q}_i^H \hat{\mathbf{s}}_i \quad \text{avec} \quad i \in [1, N_t], \tag{10}$$

où $\mathbf{s}_i = [\hat{s_1}...\hat{s}_{i-1} \quad 0 \quad \hat{s}_{i+1}...\hat{s}_{N_t}]$ représente le vecteur estimé durant l'itération précédente.

Les filtres $\mathbf{p}_i$ et $\mathbf{q}_i$ sont optimisés selon le critère MMSE et sont donnés par :

$$\mathbf{p}_i^{\text{opt}} = \sigma_s^2 \left(\mathbf{H}\mathbf{V}_i\mathbf{H}^H + \sigma_n^2\mathbf{I}_N\right)^{-1}\mathbf{h}_i, \tag{11a}$$

$$\mathbf{q}_i^{\text{opt}} = \mathbf{H}^H\mathbf{p}_i^{\text{opt}}, \tag{11b}$$

où $\mathbf{V}_i$ est une matrice diagonale représentant l'erreur résiduelle des symboles estimés.

Cet algorithme nécessite l'inversion de la matrice plusieurs fois, ce qui entraine une complexité élevée. Pour cela, plusieurs approximations sont considérées telles que MMSE-IC1 et MMSE-IC2.

Les symboles estimés $\hat{s}_i$ sont générés par le mapper pondéré à partir de l'information pondérée sur les bits du symbole qui est fournie par le décodeur de canal [26] :

$$\hat{s}_i = \mathbb{E}\{s_i\} = \sum_{s\in 2^Q} sP(s_i = s), \quad P(s_i = s) = \prod_{b=1}^{Q} P(x_{i,b}). \tag{12}$$

Ensuite, le demapper pondéré génère une information pondérée sur les bits à partir de chaque symbole égalisé [26] :

$$L(x_{i,b}) = \frac{1}{\sigma_{\eta_i}^2}\left(\min_{s\in\chi_{i,b}^{-1}} |\tilde{s}_i - \beta_i s_i|^2 - \min_{s\in\chi_{i,b}^{+1}} |\tilde{s}_i - \beta_i s_i|^2\right). \tag{13}$$

Egaliseur SIC : I-VBLAST

Le détecteur VBALST souffre du problème de propagation des erreurs. Pour cette raison, un détecteur VBLAST amélioré est proposé pour tenir compte des erreurs de décision en utilisant les symboles souples estimés. Ainsi à la première itération, ce détecteur effectue une annulation successive des symboles. Dès que les informations souples seront fournies par le décodeur de canal, les symboles estimés peuvent être alors utilisés dans les itérations suivantes [23].

Décodeurs sphériques souples

Les décodeurs sphériques sont étendus pour fournir des informations souples. Parmi ces décodeurs, on cite le décodeur de liste par sphère [27], le décodeur STS [28] et le décodeur K-Best.

Décodeur de liste par sphère

En utilisant la décomposition QR, la distance euclidienne d_1 peut être calculée recursivement à partir de la distance euclidienne cumulée d_i comme suit :

$$d_i = d_{i+1} + \underbrace{\left|\tilde{y}_i - \sum_{j=i}^{N_t} R_{i,j}s_j\right|^2}_{m_i^C} + \underbrace{\frac{N_0}{2}\sum_{b=1}^{Q}(|L_A(x_{i,b})| - x_{i,b}L_A(x_{i,b}))}_{m_i^A}, \quad i = N_t, ..., 1. \tag{14}$$

où m_i^C et m_i^A représentent respectivement le métrique de canal et le métrique *a priori*.

Le principe de décodeur de liste par sphère est de générer une liste $\mathcal{L} \subset 2^{QN_t}$ qui contient les meilleures candidats pour calculer les LLRs [27] :

$$L(x_{i,b}) = \frac{1}{N_0}\min_{\mathcal{L}\cap\chi_{i,b}^{-1}}\{d_1\} - \frac{1}{N_0}\min_{\mathcal{L}\cap\chi_{i,b}^{+1}}\{d_1\}. \tag{15}$$

Le décodeur de liste par sphère est capable d'approcher les performances limites selon la taille de la liste. Si la taille liste est assez grande, les LLRs seront calculés avec fiabilité. Par contre, si la taille est petite, il y a une possibilité de ne pas trouver quelques candidats. Ce problème est connu sous le nom de *missing counter hypotheses*. Une simple solution pour résoudre ce problème est de fixer le LLR correspondant à une valeur constante, *LLR clipping* [27].

Décodeur STS

Le LLR peut être aussi calculé comme suit :

$$L\left(x_{i,b}\right) = \begin{cases} \frac{1}{N_0}\left(d_{i,b}^{\overline{\mathrm{MAP}}} - d^{\mathrm{MAP}}\right), & \text{si } x_{i,b}^{\mathrm{MAP}} = +1 \\ \frac{1}{N_0}\left(d^{\mathrm{MAP}} - d_{i,b}^{\overline{\mathrm{MAP}}}\right), & \text{si } x_{i,b}^{\mathrm{MAP}} = -1. \end{cases}$$

$$d^{\mathrm{MAP}} = \left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}^{\mathrm{MAP}}\right\|^2 - N_0 \log P\left(\mathbf{s}^{\mathrm{MAP}}\right), \tag{16}$$

$$d_{i,b}^{\overline{\mathrm{MAP}}} = \min_{s \in \chi_{i,b}^{\overline{\mathrm{MAP}}}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 \log P\left(\mathbf{s}\right)\right\}, \tag{17}$$

$$\mathbf{s}^{\mathrm{MAP}} = \arg \min_{s \in 2^{Q \cdot N_t}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 \log P\left(\mathbf{s}\right)\right\}. \tag{18}$$

L'idée principale de décodeur STS est de calculer les LLRs simultanément en mettant à jour les métriques correspondantes (d^{MAP}, $d_{i,b}^{\overline{\mathrm{MAP}}}$) durant une seule étape de recherche [28]. Ce décodeur offre des performances optimales et peut être adapté pour fournir des décisions dures ainsi que des décisions souples selon la valeur d'écrêtage.

Décodeur K-Best proposé

Pour réduire la complexité de décodeur K-Best, nous proposons deux améliorations du décodeur K-Best classique [29]. La première amélioration simplifie l'énumération hybride des symboles dans le cas où l'information *a priori* est incorporée dans la recherche en utilisant deux LUTs. Une LUT sera utilisée pour énumérer les symboles selon les métriques de canal et l'autre LUT sera utilisée pour énumérer les symboles selon les métriques *a priori*.

La deuxième amélioration est d'utiliser une expansion à la demande des symboles pour éviter l'expansion exhaustive et les opérations de tri. Les symboles parents choisissent au début leur premier candidat, un nombre de candidats sera alors sélectionné. Si le nombre de candidats reste inférieur au nombre requis K, alors les symboles parents dont leur premier candidat est sélectionné donnent leur deuxième candidat et ainsi de suite.

Le décodeur ainsi proposé offre des performances comparables à celui du décodeur K-Best classique avec une réduction en complexité [30].

Analyse de convergence en utilisant le diagramme EXIT

Le diagramme EXIT est un outil d'analyse de convergence des systèmes itératifs. Ce diagramme permet de quantifier l'information mutuelle échangée entre deux décodeurs afin de prévoir le nombre d'itérations et le point de convergence [31]. Il est obtenu en traçant les caractéristiques, $I_E = f(I_A)$, reliant la quantité d'information mutuelle en entrée (I_A) à la quantité d'information sortante (I_E) pour chacun des décodeurs composant le récepteur itératif. La trajectoire entre les deux courbes détermine le nombre d'itérations nécessaires.

L'information mutuelle peut être déterminée par la relation approchée suivante :

$$I_x \approx 1 - \frac{1}{L_b} \sum_{n=0}^{L_b-1} \log_2 \left(1 + \exp\left(-xL_x\right)\right), \tag{19}$$

où L_b est le nombre de bits transmis, et L_x est le LLR associé avec le bit $x \in \{-1, +1\}$.

Dans le récepteur itératif considéré dans notre travail, on désigne par I_{A1} et I_{E1} les informations mutuelles à l'entrée et à la sortie de détecteur MIMO et par I_{A2} et I_{E2} les informations mutuelles de décodeur de canal (Turbo, LDPC).

Dans un premier temps, on étudie la convergence de décodage de canal notamment le décodeur turbo et le décodeur LPDC. Dans un deuxième temps, on étudie la convergence du récepteur itératif [32].

La Figure 1 représente les diagrammes EXIT de différents détecteurs et décodeurs de canal pour un système à multiplexage spatial 4×4. On peut tout d'abord noter que plus le détecteur est complexe, plus l'information mutuelle en sortie est grande. Ainsi le décodeur STS fournie une grande information en sortie que le décodeur K-Best et l'égaliseur MMSE-IC. Pour le décodeur de canal, on peut noter que plus le nombre d'itérations est grand, plus l'information mutuelle est grande. Cependant, on peut voir qu'il est inutile d'effectuer un grand nombre d'itérations à l'intérieur du décodage de canal. Ainsi, effectuer deux itérations dans le cas du décodeur turbo offre un bon compromis entre performance et complexité. Les échanges d'information sont modélisés par une trajectoire *en escalier* qui se termine au point de croisement des caractéristiques des deux décodeurs. On voit bien que quatre itérations externes sont suffisantes pour le recepteur itératif pour converger dans le cas de décodeur STS et LC-K-Best.

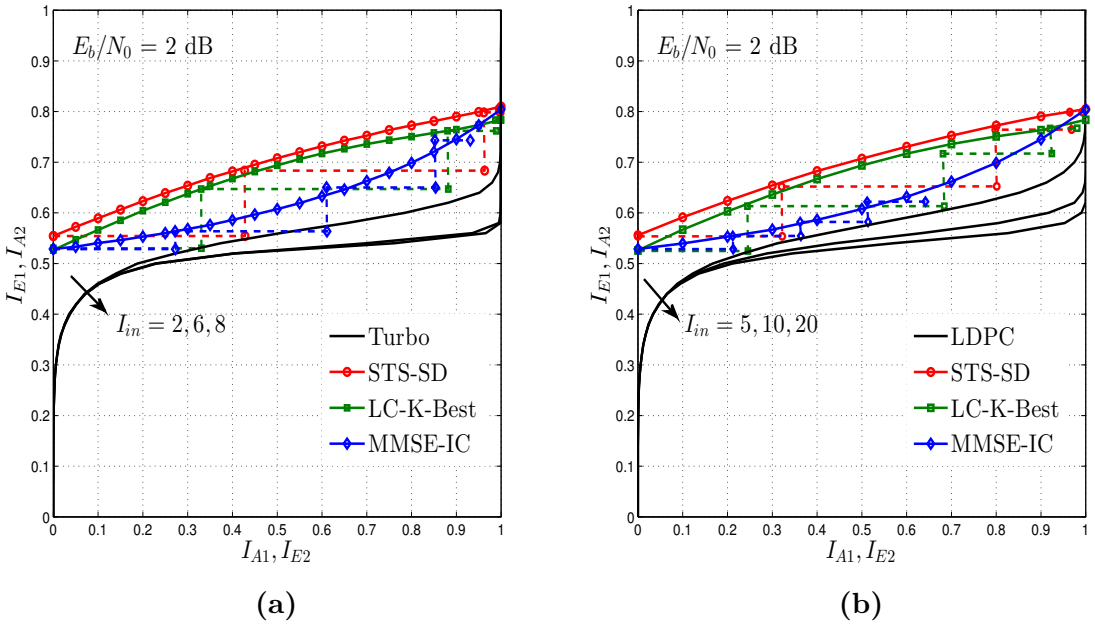
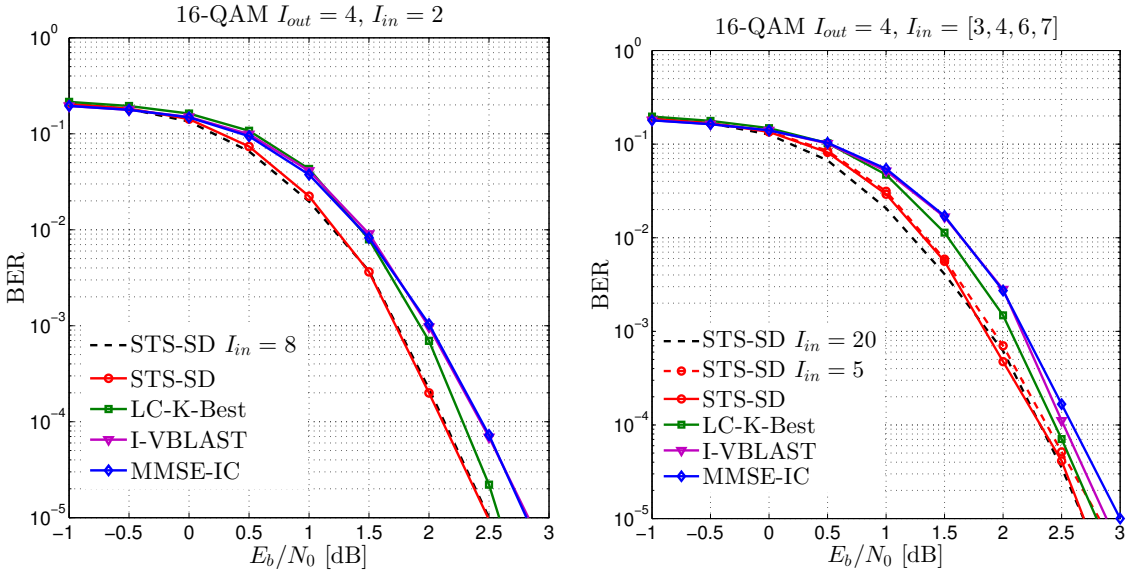


Figure 1: Diagrammes EXIT de différents détecteurs MIMO (STS-SD, LC-K-Best, MMSE-IC) et décodeurs de canal : **(a)** décodeur turbo LTE, $K_b = 1,024$, $R_c = 1/2$, et **(b)** décodeur LDPC $N_b = 1,944$, $R_c = 1/2$, à $E_b/N_0 = 2$ dB pour un système de multiplexage spatial 4×4, 16-QAM.

Performances des détecteurs

Les performances des différents détecteurs avec différents décodeurs de canal sont évaluées dans un premier temps dans un canal de Rayleigh puis dans des modèles de canaux multi-trajets. La Figure 2 présente les performances d'un système de multiplexage spatial 4×4 pour une modulation 16-QAM. On peut voir que répéter 8 itérations dans le décodeur turbo et 20 itérations dans le décodeur LDPC n'apporte pas une amélioration significative par rapport au cas où on considère un nombre total de 8 itérations ou de 20 itérations distribuées sur les 4 itérations externes. En comparant les algorithmes, on voit que le décodeur LC-K-Best présente une dégradation de moins de 0.2 dB par rapport au décodeur STS et une amélioration de 0.2 dB par rapport au égaliseur MMSE-IC et I-VBLAT. On peut voir aussi que les performances du système avec les deux décodeurs de canal, turbo et LDPC, sont comparables.



(a) LTE turbo decoder **(b)** LDPC decoder

Figure 2: Performances d'un système à multiplexage spatial 4×4 en utilisant différents détecteurs MIMO (STS-SD, LC-K-Best, I-VBLAST and MMSE-IC) dans un canal de **Rayleigh**, **(a)** décodeur turbo LTE, $K_b = 1,024$, $R_c = 1/2$, $I_{\text{out}} = 4$ $I_{\text{in}} = 2$, **(b)** décodeur LDPC $N_b = 1,944$, $R_c = 1/2$, $I_{\text{out}} = 4$ $I_{\text{in}} = [3, 4, 6, 7]$.

Ensuite, nous évaluons les performances dans des canaux plus réalistes (EPA, EVA, ETU). Des comportements équivalents sont observés. Le tableau 5 donne les valeurs de E_b/N_0 nécessaires pour atteindre un taux d'erreur binaire de 1×10^{-4} des différents détecteurs et décodeurs de canal avec plusieurs modulations. Les valeurs données dans les parenthèses correspondent à la perte de performance par rapport au décodeur STS-SD.

Chapitre 4 : Evaluation de la complexité et conversion en virgule fixe

Le chapitre 4 a pour but d'étudier les aspects d'implémentation associés au récepteur itératif telles que la complexité et la conversion en virgule fixe. Dans un premier temps, la complexité des différentes techniques de détection et de décodage de canal

Table 5: Les valeurs de $\boldsymbol{E_b/N_0}$ permettant d'atteindre un taux d'erreur binaire de $\mathbf{1 \times 10^{-4}}$ dans les modèles de canaux LTE des différents détecteurs et décodeurs de canal (turbo, LDPC) dans un système de multiplexage spatial $\mathbf{4 \times 4}$ avec une modulation 16-QAM $\boldsymbol{R_c = 1/2}$, et 64-QAM $\boldsymbol{R_c = 3/4}$.*

		décodeur turbo		décodeur LDPC	
		16-QAM	64-QAM	16-QAM	64-QAM
EPA	STS-SD	6.3 dB	14.0 dB	6.2 dB	13.8 dB
	LC-K-Best	6.5 dB (−0.2)	13.9 dB (+0.1)	6.4 dB (−0.2)	13.9 dB (−0.1)
	MMSE-IC	7.4 dB (−1.1)	> 20 dB (> −6)	8.0 dB (−1.8)	> 20 dB (> −6)
EVA	STS-SD	5.2 dB	14.3 dB	5.3 dB	13.4 dB
	LC-K-Best	5.4 dB (−0.2)	14.4 dB (−0.1)	5.6 dB (−0.3)	13.7 dB (−0.3)
	MMSE-IC	5.8 dB (−0.6)	19.0 dB (−4.7)	6.0 dB (−0.7)	18.5 dB (−5.1)
ETU	STS-SD	4.4 dB	13.0 dB	4.9 dB	12.4 dB
	LC-K-Best	4.6 dB (−0.2)	13.0 dB (0.0)	4.9 dB (0.0)	12.4dB (0.0)
	MMSE-IC	4.9 dB (−0.5)	17.5 dB (−3.5)	5.1 dB (−0.2)	16 dB (−3.6)

*La valeur en parenthèse correspond à la perte de performance par rapport au décodeur STS.

est évaluée en termes de nombre d'opérations. Nous comparons ensuite la complexité de plusieurs schémas combinant ces différentes techniques. Dans un deuxième temps, nous présentons une représentation en virgule fixe du récepteur itératif basée sur le décodeur LC-K-Best.

Complexité du récepteur itératif

La complexité d'un récepteur itératif peut être exprimée de la manière suivante [32] :

$$C_{\text{total}} = I_{\text{in}} I_{\text{out}} C_{\text{dec}} N_{\text{bit}} + N_{\text{symb}} \left\{ C_{\text{det1}} + (I_{\text{out}} - 1) C_{\text{deti}} \right\}, \tag{20}$$

où C_{det1} représente la complexité du détecteur MIMO à la première itération. Alors que C_{deti} représente la complexité aux autres itérations. C_{dec} est la complexité du décodeur du canal (turbo or LDPC). N_{bit} est le nombre de bits transmis et N_{symb} est le nombre de symboles correpondants reliés par la relation suivante :

$$N_{\text{symb}} = \frac{N_{\text{bit}}}{Q R_c N_t} = \alpha N_{\text{bit}}, \quad \text{with} \quad \alpha = \frac{1}{Q R_c N_t}, \tag{21}$$

où Q est le nombre de bits par le symbole de constellation, R_c est le rendement du code et N_t est le nombre d'antennes à l'emission.

La figure 3 montre la complexité du récepteur itératif pour un système de multiplexage spatial 4×4 avec une modulation 16-QAM. On voit que l'égaliseur MMSE-IC présente la plus faible complexité en termes de MUL et ADD. Cependant, il nécessite plusieurs opérations de DIV et de SQRT. La complexité de STS-SD est beaucoup plus élevée que LC-K-Best (60% MUL et 30%ADD). On notera aussi que la complexité du récepteur itératif avec le décodeur LDPC est plus petite que la complexité avec le décodeur turbo en termes d' ADD, Max et LUT. Cependant, les deux récepteurs présentent une complexité approximativement similaire en termes de MUL, DIV et SQRT.

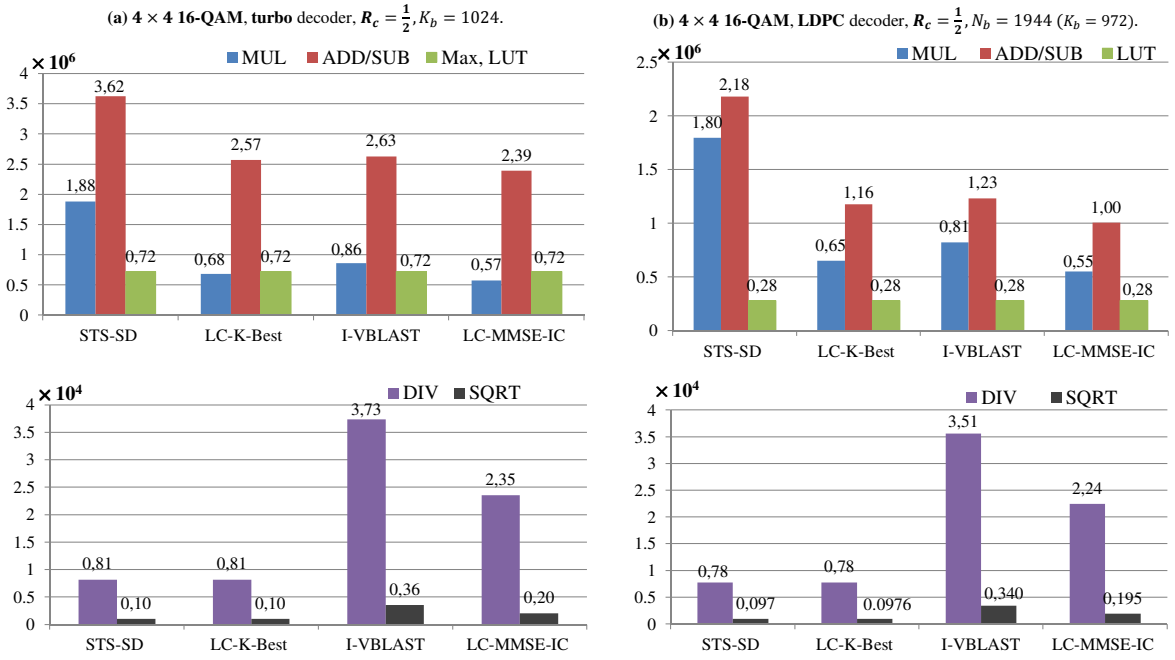


Figure 3: Complexité pour un système de multiplexage spatial **4 × 4** avec une modulation 16-QAM des différentes techniques de détection avec **(a)** décodeur turbo, et**(b)** décodeur LDPC, $R_c = 1/2$.

Représentation en virgule fixe

Pour une mise en œuvre d'une plateforme matérielle, la représentation en virgule fixe est plus efficace en termes de mémoire, de consommation d'énergie et de temps d'exécution. La conversion en virgule fixe nécessite de spécifier le nombre de bits pour la partie entière, iwl (dynamique) et pour la partie fractionnaire, iwf (précision).

Deux approches distinctes peuvent être utilisées pour la quantification des données : les approches basées sur des simulations et les approches analytiques. Les approches basées sur des simulations peuvent être appliquées à tous les types du système. Cependant, leur inconvénient majeur est le temps d'exécution assez elevé. Alors que l'approche analytique tend à trouver une expression mathématique, mais elle ne peut être appliquée que sur des systèmes linéaires. Par conséquent, dans notre travail, les approches basées sur des simulations sont utilisées. Le système est alors décomposé en plusieurs blocs, et le format des différentes variables internes, des entrées et des sorties est déminé. Le tableau 6 représente le format retenu pour les variables du système.

Les performances du récepteur itératif en virgule fixe et en virgule flottante sont comparables avec une dégradation moins de 0.2 dB avec une modulation de 64-QAM.

Chapitre 5 : Vers des systèmes MIMO-OFDM réels

Le chapitre 5 étudie plusieurs considérations en vue de l'implémentation réelle pour les systèmes MIMO-OFDM. L'estimation de canal est tout d'abord présentée. Les performances des différentes techniques d'estimation de canal pour les systèmes MIMO-OFDM sont ainsi comparées. Ensuite, la synchronisation du système MIMO-OFDM est brièvement abordée afin de pouvoir tester le récepteur itératif sur la plateforme matérielle WARP.

Table 6: Représentation des variables du système en un format fixe.

	Paramètres	(iwl, fwl)
Constellation	symbole $s(s_{\mathrm{Re}}, s_{\mathrm{Im}})$	(2,6),(2,9)
Canal	coefficients de canal h_{ij}	(3,7)
	variance de bruit N_0	(1,7)-(1,10)
décodeur turbo	LLR	(4,3)
	$\bar{\gamma}$	(5,3)
	$\bar{\alpha}$, $\bar{\beta}$	(6,3)
décodeur K-Best	$\mathbf{y}$	(4,7)
	Dist d_i	U*(5,7), U(5,9)
	$\mathbf{Q}$	(3,7), (3,9)
	$\mathbf{R}$	(4,7), (4,9)
	$\tilde{\mathbf{y}}$	(4,7), (4,9)
	LLR	(4,3)

* U réfère à la représentation en virgule fixe non signée.

Estimation du canal pour le système MIMO-OFDM

Les techniques d'estimation de canal sont classées en trois catégories : les techniques supervisées, les techniques aveugles et les techniques semi-aveugles [33].

Les techniques supervisées sont basées sur l'insertion dans la trame des symboles pilotes ou d'une séquence d'apprentissage, *training sequence.* Ces techniques sont simples à mettre en œuvre mais souffrent d'une perte d'efficacité spectrale. Alors que les techniques aveugles sont basées sur la connaissance de certaines propriétés statistiques du signal reçu. Ces techniques ont une bonne efficacité spectrale mais elles sont complexes à mettre en œuvre et ont un long temps de convergence. Les techniques semi-aveugles utilisent à la fois les symboles pilotes et les symboles de données pour faire l'estimation. Nous nous concentrons donc dans la suite du document sur les techniques supervisées car elles offrent un bon compromis entre performance et complexité.

Techniques d'estimation de canal supervisées

Dans un système MIMO-OFDM, les pilotes doivent être insérés en prenant en considération la sélectivité fréquentielle, temporelle ainsi que le nombre d'antennes. Nous considérons la structure utilisée dans le système LTE dans laquelle l'orthogonalité entre les différentes antennes est assurée par l'insertion des symboles nuls pour éviter toute interférence co-antenne. Différents estimateurs sont ainsi considérés comme l'estimateur LS, LMMSE [34, 35].

L'estimation **LS** est basée sur le critère des moindres carrés :

$$\hat{h}_{ij,k}^{\mathrm{LS}} = \frac{y_{i,k}}{s_{j,k}}, \quad k = 0, ..., N_p - 1. \tag{22}$$

où k représente l'incide de sous-porteuse, $y_{i,k}$ est le signal reçu par la $i^{\text{ème}}$ antenne et $s_{j,k}$ est le signal transmis par la $j^{\text{ème}}$ antenne. N_p correspond au nombre de porteuses pilotes. Une interpolation est ensuite effectuée pour trouver les coefficients du canal sur toutes les sous-porteuses. Plusieurs types d'interpolation existent comme l'interpolation constante, linéaire ou polynomiale. L'estimation LS est la plus simple

à mettre en œuvre mais souffre d'une forte sensibilité au bruit.

L'estimateur **LMMSE**, connu aussi sous le nom d'un filtre de Winner à deux dimensions [36], tient compte de la variance du bruit et de la corrélation temporelle et fréquentielle du canal pour améliorer et interpoler les estimés du canal au niveau des symboles pilotes. Il consiste à appliquer un filtre qui minimise l'erreur quadratique moyenne entre les coefficients parfaits de canal $\mathbf{H}$ et les estimés $\hat{\mathbf{H}}$:

$$\hat{\mathbf{H}}^{\text{LMMSE}} = \arg\min \mathbb{E}\left\{\left\|\mathbf{H} - \hat{\mathbf{H}}\right\|^2\right\}. \tag{23}$$

$$\hat{\mathbf{H}}^{\text{LMMSE}} = \mathbf{R}_{HH_p}\left(\mathbf{R}_{H_pH_P} + \sigma_n^2\mathbf{I}_{N_p}\right)^{-1}\hat{\mathbf{H}}_P^{\text{LS}}, \tag{24}$$

où $\mathbf{R}_{HH_p}$ est la matrice de cross-corrélation entre tous les sous-porteuses et les porteuses pilotes. $\mathbf{R}_{H_pH_P}$ est la matrice d'autocorrélation antre les porteuses pilotes.

L'estimateur LMMSE présente une grande complexité due à l'inversion matricielle. En plus la connaissance des matrices de corrélations fréquentielle et temporelle du canal est nécessaire. Pour cela, une simplification de cet algorithme basé sur une décomposition en valeurs singulières (SVD) est proposée. D'autres techniques basées sur le passage dans un domaine de transfert comme la DFT et DCT sont aussi proposées pour réduire en plus la complexité [37, 35, 38].

L'estimateur **DFT** par le passage dans le domaine temporel est capable d'améliorer l'estimation de canal en réduisant considérablement la puissance de bruit dans le cas où tous les sous-porteuses sont modulées. Par contre, lorsque un ensemble de sous-porteuses nulles est inséré sur les bords du spectre, le MSE de l'estimateur (DFT) par passage dans le domaine temporel présente de fortes discontinuités, ce qui entraine une dégradation des performances du système [35]. Pour franchir ce problème, une approche basée sur l'utilisation de pseudo-inverse avec une décomposition en valeur singulière (SVD) tronquée est ainsi considérée (TSVD) [39, 40]. Cette technique consiste à utiliser un seuil et à considérer seulement les valeurs singulières supérieures à ce seuil pour avoir une matrice bien conditionnée réduisant ainsi la sensibilité au bruit. La valeur du seuil influe directement sur les performances de l'estimateur. Une valeur fixée à 10% de la valeur singulière est utilisée dans notre travail. La Figure 4 représente les performances des techniques d'estimation de canal pour un multiplexage spatial 4×4 avec une modulation 16-QAM dans différents types de canaux. On voit bien que la technique TSVD présente des performances proches de l'estimateur LMMSE avec une dégradation de 0.5 dans le cas où le canal est peu variant (EPA). Par contre, lorsque le canal est sélectif en temps et en fréquence (EVA), ces techniques présentent une forte dégradation. D'autres techniques plus avancées doivent être donc utilisées pour pouvoir suivre la variation du canal.

Synchronisation du système MIMO-OFDM

Dans les systèmes de transmission réels, le récepteur ne connait pas le début et la fin de la trame ainsi que la fréquence d'échantillonnage de l'émetteur. Une synchronisation temporelle et fréquentielle doit être donc effectuée pour éviter la perte d'orthogonalité des sous-porteuses et la dégradation des performances du système [41].

La synchronisation est généralement décomposée en deux étapes : une synchronisation grossière et une synchronisation fine. La synchronisation temporelle consiste à déterminer le début exact de la trame. Alors que la synchronisation fréquentielle consiste à corriger le décalage de la fréquence porteuse (CFO) entre l'émetteur et le

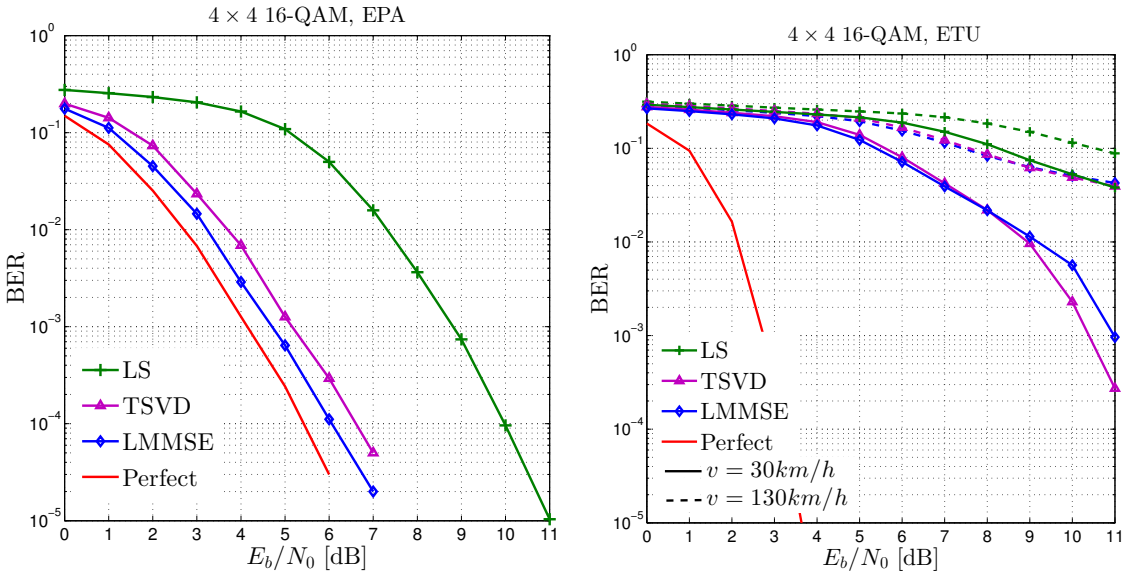


Figure 4: Performance d'un système de multiplexage spatial $\mathbf{4 \times 4}$ avec une modulation 16-QAM en utilisant différentes techniques d'estimation de canal (LS, LMMSE, TSVD), **(a)** EPA, et **(b)** ETU, $N = 1024$, $N_c = 600$.

récepteur. Dans notre système, on utilise un même oscillateur local, le décalage de la fréquence est donc nul. Pour la synchronisation temporelle, des séquences CASAC orthogonaux sont utilisées. La corrélation entre ces séquences et le signal reçu est ainsi effectuée pour trouver le pic qui correspond au début de la trame.

Plateforme d'essai du système MIMO-OFDM dans un environnement réel

La plateforme matérielle WARP est utilisée pour tester les performances du récepteur proposé dans un environnement réel. Le signal émis ainsi que la bande occupée sont visualisés. Ensuite, différentes configurations des antennes et différents nombres d'itérations sont utilisés pour valider le schéma proposé.

Conclusions et perspectives

Le récepteur itératif est largement considéré car il est capable d'approcher les performances limites dans les systèmes de communication sans fil. Cependant, ce récepteur est difficile à mettre en œuvre. Cette thèse est consacrée à étudier la convergence, la performance et la complexité du récepteur itératif pour un système MIMO-OFDM afin de trouver un bon compromis entre performance et complexité.

Nous avons introduit au début les différentes techniques de détection et de décodage de canal. Un décodeur MIMO basé sur l'algorithme K-Best est ainsi proposé qui réduit significativement la complexité du décodeur sphérique. La convergence du récepteur itératif a été ensuite analysée en utilisant le diagramme EXIT pour trouver le nombre d'itérations internes et externes. Puis, les performances du récepteur proposé ont été présentées et comparées avec différentes ordres de modulation et types de canaux.

En outre, la complexité du récepteur itératif avec les différentes techniques de détection et de décodage de canal est étudiée et comparée avec différentes modulations. Ensuite, une représentation en virgule fixe des paramètres internes et des entrées et des sorties a été introduite.

Nous avons également rappelé les différentes techniques d'estimation de canal pour le système MIMO-OFDM. Une brève description de la synchronisation a été encore présentée. Finalement, le récepteur itératif a été testé sur la plateforme matérielle WARP validant la performance dans un environnement réel.

D'autres études peuvent également être menées sur la réception itérative. L'optimisation de la complexité du récepteur proposé et sa mise en œuvre peuvent être investiguées. L'estimation de la consommation d'énergie et l'estimation de la latence peuvent être aussi envisagées. De nouvelles techniques d'estimation de canal constituent aussi un sujet d'intérêt.

Introduction

Contents

Communication systems have known a big evolution over the last decade. Their developments have been driven by the increase of human demand to get information with better quality of service in shortest possible time and at highest speed. However, many challenges arise and are directly related to the limited transmission power, frequency spectrum allocation and channel propagation issues as time and frequency fading. Nowadays advanced techniques have been developed in order to achieve reliable high data rate requirements for future wireless communication systems. These techniques include multiple-input multiple-output (MIMO) technique, multi-carriers technique and channel coding techniques (i.e., turbo codes, LDPC codes). These techniques have been adopted into the evolving wireless standards such as 3GPP-LTE (A) for cellular networks, IEEE 802.11n/ac and 802.16e/m for wireless local (WLAN) and wide area networks (WiMax) and DVB-RCS, DVB-T2 for digital video broadcasting.

MIMO technology which utilizes multiple antennas at the transmitter and/or receiver side is able to achieve high diversity through space time coding and high data rate through spatial multiplexing without the need of additional spectrum and transmit power. On the other hand, orthogonal frequency division multiplexing (OFDM) is a well suitable technology to combat inter-symbol interference and therefore achieve better spectral efficiency. Indeed, OFDM transforms a frequency-selective channel into non-selective sub-channels by dividing the available spectrum into a number of orthogonal sub-channels. A cyclic prefix is then added to further eliminate the inter-symbol interferences. Whereas, modern channel coding schemes such as turbo codes or LDPC codes are powerful forward error correction (FEC) codes able to protect the integrity of the transmitted data and to approach the channel capacity.

Therefore, the combination of MIMO-OFDM with channel coding has been seen an attractive solution for future high data rate and reliable transmissions. However, the practical design of coded MIMO-OFDM systems involves numerous challenges at the receiver, especially for MIMO schemes introducing interference. For coded MIMO-OFDM systems, a linear superposition of transmitted symbols is observed at the receiver. Therefore, an advanced receiver must be used to recover the transmitted symbols from interferences.

The reception strategy that offers best performance is to jointly detect and decode the received symbols [42]. However, this joint detection scheme has been shown to be very complex and infeasible for practical implementation. Alternately, the optimal performance can be approached by the iterative processing or commonly referred to as turbo processing which replaces the joint detection by iteratively performing independent detection and decoding processing. It consists of soft-input soft-output (SISO) detector and channel decoder that exchange soft information. In spite of excellent error rate performance, such an iterative receiver exhibits a high computational complexity, which depends on MIMO detection algorithms, channel decoding algorithms and number of iterations. Therefore, the complexity reduction of such an iterative receiver becomes a major research issue. Since in most applications, the channel decoder is often a convolutional decoder, a turbo decoder or an LDPC decoder, an efficient implementation of MIMO detection algorithms and an optimization of the number of iterations have to be investigated for a good trade-off between performance and complexity.

Regarding the MIMO detection method, the optimal detection relies on maximum a posteriori probability (MAP) algorithm. However, it presents a complexity that exponentially increases with respect to the number of transmit antennas and the modulation order. Therefore the design of sub-optimal detection algorithms has known an intense research in the literature ranging from hard-decision detectors to soft-input soft-output detectors with different levels of complexity and performance. While the hard-decision detectors deliver binary estimates of transmitted symbol vectors, the soft-output detectors provide soft estimates in form of log likelihood ratios providing more reliability but with an increase of computational complexity.

Besides, the transmitted information is distorted by the radio channel which is unknown at the receiver. Hence channel must be accurately estimated in order to recover the transmitted signal. Channel estimation is another challenging problem in MIMO-OFDM systems. Unlike the single carrier and single antenna systems which require to find only one channel, MIMO-OFDM systems required to find multiples channels for each subcarrier and for each link between transmit and receive antennas. Generally, the channel could be estimated using a preamble or pilot symbols known at both transmitter and receiver. Several channel estimation techniques have been widely developed in the literature. Yet, various aspects have to be considered in channel estimation such as the required performance, the complexity and time variation of the channel.

Researches first focused on the theoretical aspects of MIMO systems by evaluating the theoretical performance and proposing an optimal receiver. However the researches today focus on the implementation aspects of such systems in real environments. The implementation aspects involve in turn numerous challenges about fixed point representation, computational complexity, real throughput, latency and power consumption. The conversion into a fixed-point format requires to specify a finite word length (wl) with a specific number of bits for integer parts (iwl) and for fractional parts (fwl). However, this conversion has a great impact on system performance owing to the reduced dynamic range and the precision inaccuracy.

Additionally, in real communication systems, the synchronization of MIMO-OFDM system is a key issue. The synchronization is required in both time and frequency. The time synchronization involves finding the start of the received frame. Meanwhile, the

frequency synchronization deals of finding a frequency offset estimation between the transmitter and the receiver local oscillators.
The research activity about iterative processing for MIMO-OFDM systems is still very much evolving in order to find a best trade off between performance and complexity for the future wireless generation.

Objectives and Contributions

The main objective of this thesis is to address the above challenges involved by the iterative receiver combining MIMO detection with channel decoding. Therefore, an advanced receiver must be developed at algorithmic and architectural levels that must be able to achieve near optimal performance with an acceptable computational complexity. The receiver must also satisfy high throughput, low latency and low power consumption requirements for wireless communication systems. Moreover, the receiver architecture has to be suitable for parallel implementation, scalable and configurable to support different parameters and applications.

The main contributions of this thesis can be summarized as follows:

- Analysis of the main existing soft-input soft-output detection algorithms for iterative receivers. Consequently, a low complexity K-Best decoder (LC-K-Best) is proposed for the sake of low computational complexity and low latency without significant performance degradation.
- Investigation of the convergence behavior of iterative receivers in order to find the required number of inner/outer iterations. The extrinsic information transfer (EXIT) charts are adopted for a thorough analysis of the convergence behaviors of the system. Consequently, a new scheduling of the number of iterations is proposed for an efficient trade-off between performance and complexity.
- Detailed analysis of the computational complexity of the iterative receivers in terms of floating point operations.
- A comparative study in terms of performance and complexity of LC-K-Best decoder and main existing MIMO detection algorithms with different configurations, modulation orders and LTE channel models. Simulation results reveal that the proposed LC-K-Best decoder achieves the best trade-off between performance and complexity among existing solutions.
- Proposition of an efficient fixed-point arithmetic of iterative receiver based on LC-K-Best decoder in order to reduce the hardware costs in terms of area, power consumption and execution time. Simulation results show that fixed-point representation almost achieves the same bit error rate (BER) performance as floating point performance.
- Analysis of the impact of imperfect channel estimation on the performance of MIMO-OFDM system. Various channel estimation techniques are therefore compared to efficiently estimate the channels.
- Development of a testbed for MIMO-OFDM system using the WARP platform in order to validate the performance of our proposed approach in real-time environments. To this end, the impact of the front-end radio frequency (RF) (i.e., amplifier, filter, ADC, DAC), the channel estimation and the time synchronization are investigated.

Thesis outline

This thesis manuscript has been split into five distinct chapters.

Chapter 1 presents the fundamental concepts of wireless communication systems useful for the rest of the work. Wireless communication systems as well as channel models and channel characteristics are first presented. Channel coding techniques namely turbo codes and LDPC codes are then reviewed. Furthermore, we present OFDM and MIMO techniques including the channel capacity, the gain and the transmission schemes. Finally, the system model used in the remainder of this thesis and the detection problematic are introduced.

Chapter 2 provides a review of main existing hard-decision MIMO detection methods proposed in the literature. As a baseline, we start by looking at the optimal maximum likelihood (ML) detection of uncoded transmission. We then discuss suboptimal methods namely linear equalizers, interference cancellation detectors and tree search detectors. Their advantages, their limitations and their performances are hence discussed. We show that the hard-decision MIMO detectors result in poor performance in terms of BER which can be improved through the use of channel coding and more advanced receivers as will be discussed on the next chapter.

In **Chapter 3**, the focus is shifted to coded transmission in which iterative MIMO receiver is investigated. The concept of iterative detection-decoding process is first illustrated followed by a description of the main existing soft-input soft-output MIMO detection. A low complexity K-Best decoder (LC-K-Best) is consequently proposed to reduce the computational complexity of the receiver. In addition, the convergence behavior of the iterative receiver is analyzed using EXIT charts. Based on this analysis, we retrieve the required number of inner and outer iterations for the convergence of the iterative receiver when using turbo and LDPC decoders. The performance of these systems is then compared and discussed with different configurations, modulation orders and various LTE channel models.

Chapter 4 investigates the computational complexity of the iterative receiver and its conversion into an efficient fixed-point arithmetic. First, the complexity of the iterative receiver is presented. Next, the complexity of different detection algorithms are discussed. Furthermore, the complexity of turbo decoder and LDPC decoder is presented. A comparative study of the complexity of different configurations is hence carried out. Secondly, fixed-point conversion of the receiver is conducted for an efficient implementation. We hence describe the approach used in order to find the required word length of the input/output and intermediate signals. Consequently, an efficient fixed-point format of system parameters for the iterative receiver is presented.

Chapter 5 is devoted to study the impact of real-time considerations namely channel estimation and time synchronization on the performance of MIMO-OFDM systems. Several channel estimation methods in frequency and time domain are therefore reviewed. Furthermore, we compare the performance of these techniques in different channel environments. Next, the synchronization of MIMO-OFDM system is briefly presented. The proposed iterative receiver is tested in a real environment using the WARP platform.

Finally the conclusions and future works drawn from the research are presented.

Chapter 1

Preliminaries

Contents

This chapter gives an overview of the basic concepts used in this thesis. Wireless communication systems are first introduced. We then present the channel models and the main characteristics of multipath channels. Next, channel coding including turbo codes and LDPC codes used in the sequel of the thesis are described. Afterward, we illustrate the bases of OFDM transmission schemes. Following that, a review is done about the existing MIMO transmission schemes, the gain and the capacity achieved by using multiple antennas. Finally, we present MIMO system model and the detection and decoding problems which will constitute the motivation and stimulation of the research carried out in this thesis.

1.1 Wireless communication systems

The aim of digital communication systems is to transmit the information with highest reliability. The general architecture of such a system consists of three basic components: the transmitter, the wireless channel and the receiver as illustrated in Figure 1.1.

At the transmitter side, the source sends the information to the source encoder. The source encoder converts this signal into a binary data stream $\mathbf{u}$ of length K_b as short as possible to reduce redundancy and consequently to reduce the bandwidth requirement of the system.

The bits stream $\mathbf{u}$ is encoded by a channel encoder which outputs a codeword $\mathbf{c}$ of length N_b. The channel encoder introduces redundancy to the data to make it more

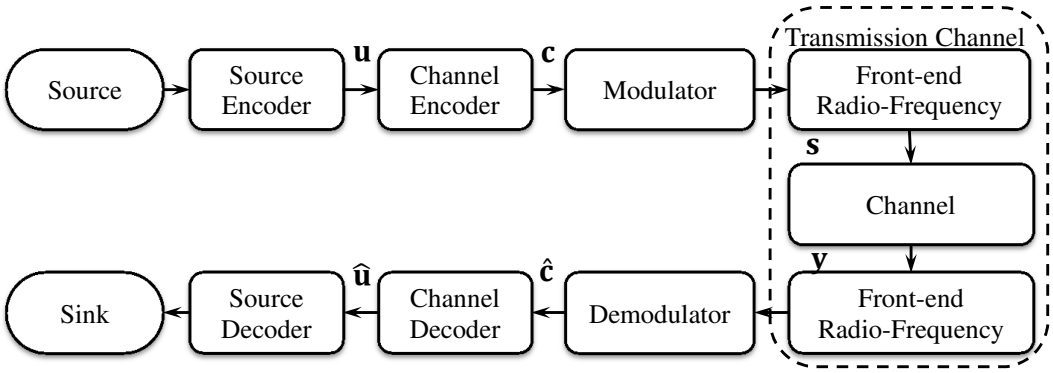


Figure 1.1: Block diagram of digital communication system.

reliable to control and correct the errors caused by noise or interference through the transmission channel. The channel coding rate is defined as $R_c = K_b/N_b$.

The coded data $\mathbf{c}$ is then mapped into a carrier signal $\mathbf{s}$ through the modulator. Thus, if we consider a M_c-ary modulation, digital modulation maps $Q = \log_2(M_c)$ bits to one of the M_c possible coded symbols. The front-end radio-frequency includes many modules such as filter, amplifier and antennas.

Then the signal passes through radio propagation channel where many perturbations may affect the signal such as noise, reflexion, diffraction, Doppler effect and multipath fading. Different channel models have been proposed in the literature to represent these variations.

At the receiver side, a corrupted version $\mathbf{y}$ of the transmitted signal $\mathbf{s}$ is received. The reverse signal processing happens. The demodulator and the decoder try to recover the original signal and generate an estimate of transmitted information $\hat{\mathbf{u}}$ and send it to the destination (sink).

1.2 Channel models

The channel refers to the medium between the transmitter and the receiver. An efficient channel model is required for the analysis, design and deployment of wireless communication systems.

The fundamental channel model is the additive white Gaussian noise (AWGN) channel. It is a memoryless channel free of interference, dispersion and fading. In this model, the signal is only corrupted by an additive Gaussian-distributed random noise with constant spectral density, $n_k \sim \mathcal{CN}(0, \sigma_n^2)$:

$$y_k = s_k + n_k. \tag{1.1}$$

Rayleigh flat fading is another theoretical channel model. Such a model assumes that the channel independently varies between successive symbols, where the channel coefficient h_k is a zero mean complex Gaussian process, $h_k \sim \mathcal{CN}(0, 1)$. Its magnitude $|h_k|$ is Rayleigh distributed,

$$y_k = h_k s_k + n_k. \tag{1.2}$$

In practice, this transmission channel requires the presence of an interleaver whose depth exceeds the coherence time of the fading process.

However, the real wireless channel is more complicated due to complex propagation environment. The transmitted signal may be reflected, diffracted and scattered by surrounding obstacles, or attenuated and absorbed by obstructions causing a fluctuation of signal power. Therefore, the signal attenuation is related to the variation of the channel strength over the time and frequency commonly referred to as fading phenomenon. Figure 1.2 classifies the different types of fading channels.

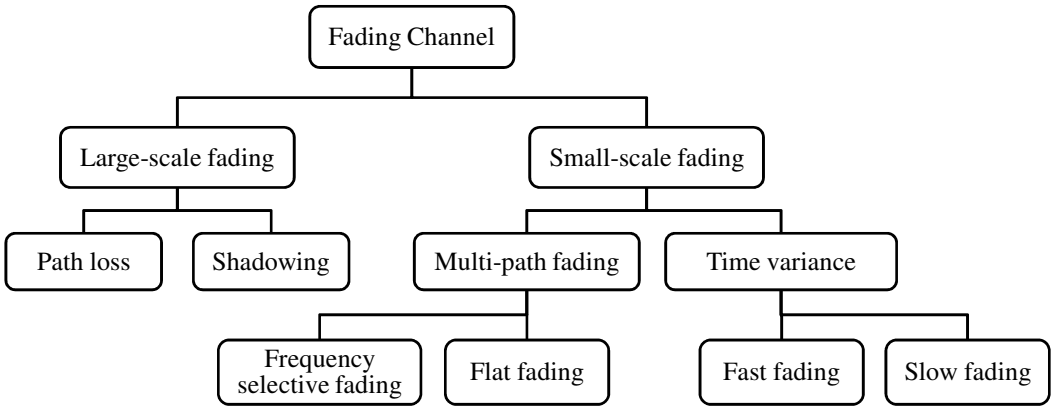


Figure 1.2: Classification of fading channels.

The fading phenomenon can be classified into two different types: large-scale fading and small-scale fading as illustrated in Figure 1.3.

1.2.1 Large-scale fading

Large-scale fading occurs as the mobile moves through a large distance. It represents the *path loss* due to the propagation. It is proportional to $1/d^{\alpha}$ where d is the distance between the transmitter and the receiver, α is the path loss exponent varying from 2 in free space propagation to 5 in urban environment.

In addition, the presence of obstructions such as buildings, mountains and walls causes more or less strong attenuation of the transmitted signal. This is known as the *shadowing* effect and it can be described by a log-normal distribution χ with standard deviation σ. The overall path loss $P_L(d)$ can be expressed by:

$$P_L(d)\,[dB] = P_L(d_0) + 10\alpha \log_{10}(\frac{d_0}{d}) + \chi_\sigma, \tag{1.3}$$

where $P_L(d_0)$ is the mean path loss in dB at a reference distance d_0 from the transmitter, α is the path loss exponent, and χ_σ is a random variable with zero mean and standard deviation σ. χ_σ typically ranges from 6 to 10 dB.

1.2.2 Small-scale fading

In contrast to large-scale fading, small-scale fading is due to rapid variations in the received signal over short distance or short time intervals. It is mainly caused by multipath propagation and mobile speed.

Depending on the delay spread, the channel can be frequency-selective or frequency flat channel. Meanwhile, depending on the time variation due to mobile speed (Doppler spread), the channel can be either fast fading or slow fading as shown in Figure 1.2.

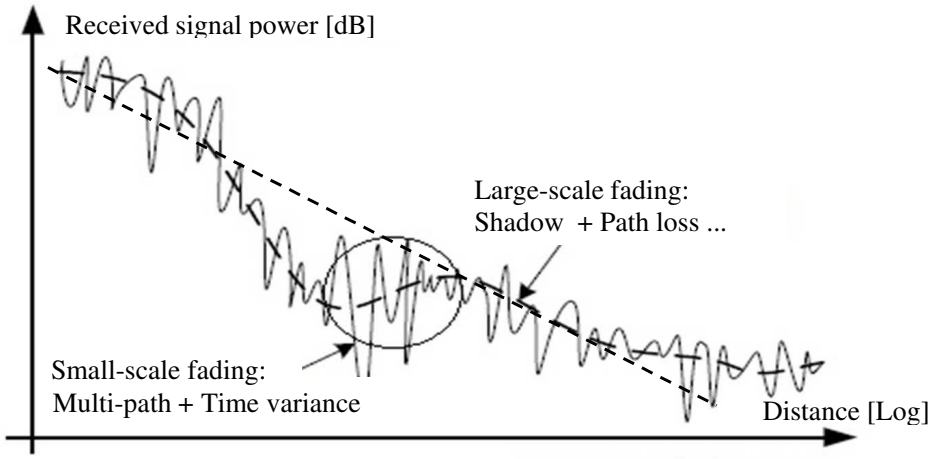


Figure 1.3: Large-scale vs small-scale fading.

Multipath propagation is the consequence of reflexion, diffraction and scattering of the transmitted signal. Thus multiple signals may arrive at the receiver from different paths with different delays, amplitudes and phases and superpose either in a constructive or destructive way. A maximum delay spread $\tau_{\max}$ is defined as the difference between the longest and shortest path with significant energy.

Doppler spread is defined by the frequency dispersion due to relative motion of the receiver or the transmitter or even the motion of the elements in the propagation environment. This results in a variation of the channel impulse response. The Doppler frequency is defined by:

$$f_d = f_m \cos\theta, \quad f_m = \frac{v f_c}{c}, \tag{1.4}$$

where f_m is the maximum Doppler frequency, f_c is the frequency of the propagated signal, c is the speed of light and v is the relative motion between the transmitter and the receiver. θ denotes the arrival angle between the incident signal and the velocity vector.

1.2.3 Multipath channel models

Consider a single-input single-output system, the channel can be represented as a linear filter with a time variant impulse response $h(t,\tau)$. The received signal $y(t)$ at time t can be described through the relation:

$$y(t) = s(t) * h(t,\tau) + n(t) = \int_{-\infty}^{+\infty} h(t,\tau) s(t-\tau) d\tau + n(t), \tag{1.5}$$

where $s(t)$ is the transmitted signal, $n(t)$ is the corrupting noise and $*$ denotes the convolution operation.

The channel impulse response (CIR) $h(t,\tau)$ for the multipath fading channel is given by:

$$h(t,\tau) = \sum_{l} \alpha_l(t)\delta(\tau - \tau_l(t)), \tag{1.6}$$

where $\alpha_l(t)$ and $\tau_l(t)$ are respectively the attenuation and the delay of the l^{th} propagation path at time t.

The characteristic of the channel can also be represented in the frequency domain by the channel frequency response (CFR). CFR can be obtained by applying a Fourier transform to the CIR in the time-delay domain:

$$\begin{aligned} H(f,t) &= \int_{-\infty}^{+\infty} h(t,\tau) \exp\left\{-j2\pi f\tau\right\} d\tau \\ &= \sum_{l} \alpha_l(t) \exp\left\{-j2\pi f\tau_l(t)\right\}. \end{aligned} \tag{1.7}$$

The continuous-time system model is generally sampled into an equivalent discrete-time model. The received signal can be expressed as:

$$y_k = \sum_{l=0}^{L-1} h_{l,k} s_{k-l} + n_k, \tag{1.8}$$

where $y_k = y(kT_e)$, k is an integer and T_e is the sampling period. L is the number of taps, $h_{l,k}$ is the l^{th} channel tap. The frequency domain response becomes:

$$H(f,k) = \sum_{l=0}^{L-1} h_{l,k} \exp\left\{-j2\pi l f T_s\right\}. \tag{1.9}$$

The signal to noise SNR ratio at the input of the receiver is defined by:

$$\text{SNR} = \frac{\mathbb{E}\left\{\left|\sum_{l=0}^{L-1} h_{l,k} s_{k-l}\right|^2\right\}}{\mathbb{E}\left\{\left|n_k\right|^2\right\}} = \frac{\left\|h_k\right\|^2 \sigma_s^2}{\sigma_n^2}. \tag{1.10}$$

Delay spread and coherence bandwidth

The coherence bandwidth B_c is defined by the frequency band over which the channel response is correlated and has a constant amplitude and a linear phase. It is inversely proportional to the root mean square (RMS) delay spread of the channel τ_{rms} [43, 1]:

$$B_c \approx \frac{1}{\tau_{\text{rms}}}. \tag{1.11}$$

The expression of B_c in equation (1.11) depends on the chosen frequency correlation coefficient ρ_f [1]:

$$B_c \approx \frac{1}{5\tau_{\text{rms}}} \quad (\text{for} \quad \rho_f = 0.5) \quad \text{or} \quad B_c \approx \frac{1}{50\tau_{\text{rms}}} \quad (\text{for} \quad \rho_f = 0.9). \tag{1.12}$$

Due to time dispersion and multipath propagation, the channel can be categorized as frequency flat fading or frequency selective fading.

Frequency flat fading: A channel is referred to as frequency flat if the coherence bandwidth B_c is greater than the bandwidth B_s of the transmitted signal. All frequency components of the signal will experience the same amount of fading.

$$B_s \ll B_c, \quad T_s \gg \tau_{\text{rms}}. \tag{1.13}$$

Frequency selective fading: A channel is frequency selective if B_s is greater than the coherence bandwidth B_c. The delay spread is much greater than the symbol period.

$$B_s > B_c, \quad T_s < \tau_{\text{rms}}. \tag{1.14}$$

In this case, different frequency components will undergo different amount of fading leading to inter symbol interference (ISI). To deal with ISI, various equalization techniques were proposed. In the rest of this thesis, a flat fading channel is assumed through the use of OFDM techniques.

Doppler spread and coherence time

The channel can be also time selective due to Doppler effect. The channel coherence time T_c is defined by the duration over which the channel impulse response can be considered constant. It is often approximated by to the inverse of Doppler frequency [44, 1]:

$$T_c \approx \frac{1}{f_m}. \tag{1.15}$$

For a correlation coefficient of 0.5 or above, the coherence time can be expressed by [1]:

$$T_c \approx \frac{9}{16\pi f_m}, \quad T_c \approx \sqrt{\left(\frac{1}{f_m}\right)\left(\frac{9}{16\pi f_m}\right)} = \frac{0.4231}{f_m}. \tag{1.16}$$

If the symbol period T_s is less than the coherence time, the channel is non time-selective (slow fading); and inversely if $T_s > T_c$ the channel is time-selective (fast fading channel). The channel can be also quasi static if it remains constant during the transmission of a frame, or block fading if it remains constant for a given number of subblocks.

We note that in case of MIMO systems, in addition to the time and frequency dispersion of the channel, another aspect must be taken into consideration: the spatial dispersion. Hence spatial correlation must be evaluated due to spatial scattering conditions resulting from the use of multiple antennas.

Different channel models have been developed for different propagation conditions. These models range from geometrical to stochastic models and from theoretical to realistic models. They modulate wireless channel in different scenarios indoor, outdoor, LOS, NLOS.

1.3 Channel coding

Channel coding or forward error correction is used in order to transmit the information with high reliability and low probability of error over noisy channels. Its principle consists in introducing redundancy into the transmitted information in order to enable the receiver to detect and correct the errors.

In 1948, Shannon [45] introduced the channel coding theorem and defined the channel capacity as the maximum rate of information that can be reliably transmitted over a communication channel. The Shannon theorem states that for a given noisy channel with channel capacity C and transmission information rate R, if $R < C$, then there exists a code that allows the probability of error to be made arbitrarily small at the receiver. This means that it is theoretically possible to transmit information nearly without error at any rate below the limiting rate C.

In case of an AWGN memory less channel, the channel capacity is given by:

$$C = B_s . \log_2 (1 + SNR) , \tag{1.17}$$

where SNR is the signal to noise ratio and B_s denotes the signal bandwidth.

Searching for practical coding schemes that achieve performance close to the Shannon bound has kept the coding theory community busy for over 50 years. Therefore, different kinds of channel codes have been developed for wireless communication systems such as convolutional codes [46], low density parity check codes (LDPC) [3], turbo codes [2] and Reed-Solomon codes.

LDPC codes were largely forgotten after their discovery for about 30 years until the development of turbo codes in 1993 [2], and have known a revisited interest in the design of powerful coding schemes [47]. LDPC codes are nowadays adopted in many emerging wireless standards namely IEEE 802.11 and DVB-T2. Whereas, binary turbo codes are included in LTE-(A) standard as a powerful coding schemes and double binary turbo codes are used in WiMax. In the present work, turbo codes and LDPC codes are considered and they are clearly described in the sequel.

1.3.1 Turbo codes

Turbo codes were initially proposed in 1993 by Berrou et al.[2] and have attracted a lot of interest over the years, since they offer performance closer to the Shannon limit. However, this advantage comes at the expense of high computational complexity, memory requirement and decoding delay. In order to achieve high throughput decoding, several approaches for parallel turbo decoder design have been proposed. In this section, we review the turbo encoder and decoder functionalities. We then describe the interleaver and the rate matching functionalities.

1.3.1.1 Turbo encoder

Turbo encoder consists of a parallel concatenation of two recursive systematic convolutional encoders separated by an interleaver. The first encoder processes the input data while the second processes the interleaved version of data.

In this work, we consider the turbo encoder as specified in 3GPP LTE standard [48]. It consists of two recursive systematic encoders with 8 states and the polynomial generators $(13, 15)_o$ (octal form), separated by a quadratic permutation polynomial (QPP) interleaver as illustrated in Figure 1.4. For each information bit u_k, the encoder outputs a systematic bit s_k, a parity bit p_{1k} from the first component encoder and a parity bit p_{2k} from the second component encoder.

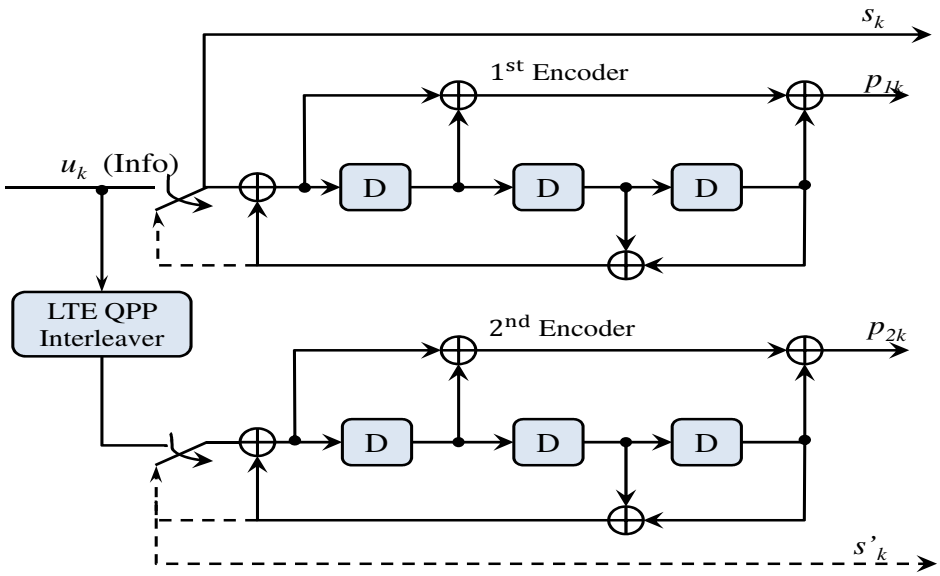


Figure 1.4: Structure of rate 1/3 LTE turbo encoder (dotted lines apply for trellis termination).

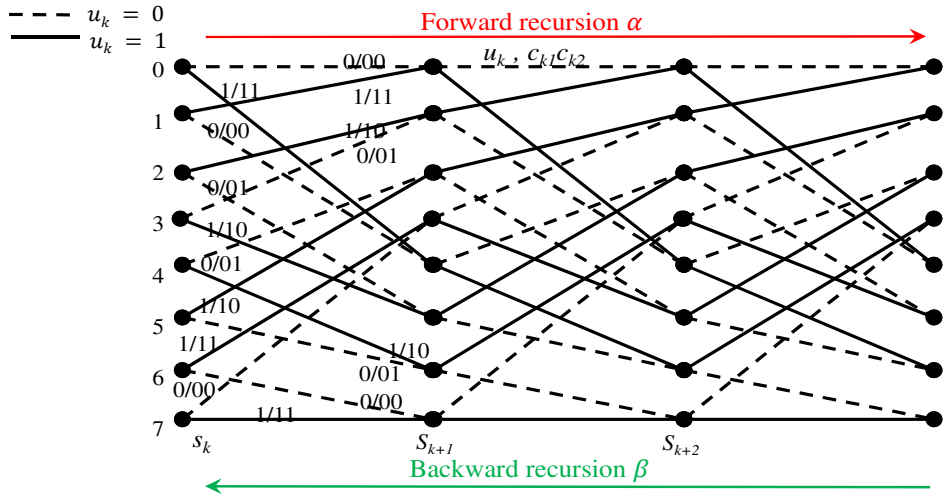


Figure 1.5: Trellis diagram of the 8-state binary LTE turbo code.

Each convolutional component encoder can be efficiently represented by a trellis diagram, which describes all possible state transitions as illustrated in Figure 1.5.

In order to force the encoder to return to its initial state, tail bits are added to terminate the trellis. This is done by taking the bits from the feedback shift register (dotted lines in Figure 1.4). However, this method presents a little loss of spectral efficiency due to the transmission of additional bits. Another method consists in adopting circular encoding [49], in which the code rate remains unchanged. Circular encoding necessitates a precoding step to determine the final state, that will be used as an initial state for the next encoding process; the code trellis can be then viewed as a circle. The first method (insertion of tail bits) is adopted in LTE standard since it is easy to implement, and for a large block size, the loss of efficiency can be considered as negligible.

In the basic form of turbo code, the initial coding rate is 1/3. To get other code

rates, rate matching technique based on a circular buffer can be used to perform either puncturing or repetition [48]. This technique is described later in this section.

1.3.1.2 Turbo decoder

The turbo decoder consists of two soft-input soft-output decoders with an interleaver and a de-interleaver between them as shown in Figure 1.6. Two families of decoding algorithms are proposed in the literature: soft-output Veterbi algorithms (SOVA) [50, 51] and maximum *a posteriori* (MAP) algorithm and its approximations [4].

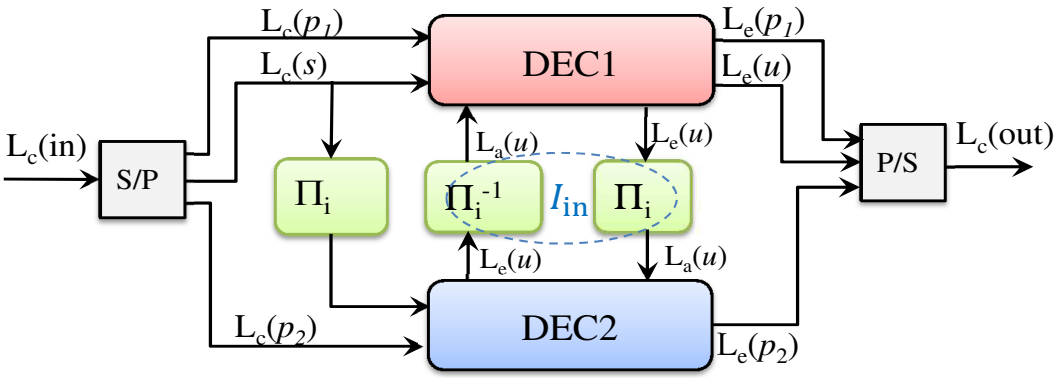


Figure 1.6: Turbo decoder.

Turbo decoder operates in an iterative process. Each full iteration consists of two half iterations, one with non-interleaved information and the second with interleaved information. The extrinsic information from one decoder is used by the other decoder as *a priori* information after interleaving or de-interleaving.

The decoder actually continues the iterative process for a specified number of iterations. However, several methods for early termination have been proposed to dynamically control the number of iterations and terminate the iterative process [52]. These methods are based on the examination of the results at each iteration to see if a sufficient decoding accuracy is achieved. One method for early termination is to view hard-decision about information bits at each half iteration and compare it with the previous values. If the information matches for a certain threshold, the iterative decoding can be then terminated.

MAP decoder

The optimal soft-input soft-output decoder for convolutional code is the BCJR (Bahl, Cocke, Jelinek and Raviv) algorithm also referred to as MAP algorithm [4]. However, due to its high complexity, the log-MAP algorithm derived in the logarithm domain and the sub-optimal max-log-MAP algorithm are practically used.
For a m-binary convolutional code, the MAP algorithm computes for each coded symbol (u_k) 2^m *a posteriori* probabilities $Pr\,(u_k = j|y)$:

$$Pr\,(u_k = j|y) = \frac{p\,(u_k = j, y)}{\sum_{i=0}^{2^m-1} p\,(u_k = i, y)}, \tag{1.18}$$

where $p(u_k = j, y)$ is the joint probability at each time k, that can be computed in the trellis structure by looking at the previous state ($\alpha_{k-1}(s_{k-1})$), next state ($\beta_k(s_k)$) and the probability of transition between these two states s_{k-1}, s_k ($\gamma_k(s_{k-1}, s_k)$).

$$p(u_k = j, y) = \sum_{(s_{k-1}, s_k)/u_k = j} \alpha_{k-1}(s_{k-1}) \beta_k(s_k) \gamma_k(s_{k-1}, s_k). \tag{1.19}$$

Therefore, the BCJR algorithm can be decomposed into 4 principal steps:

- Branch metrics $\gamma_k(s_{k-1}, s_k)$ which represent the transition probability between two states s_{k-1} and s_k at the instant k.
- Forward state metrics $\alpha_k(s_k)$ which correspond to the probability of the trellis to be in state s_k at the instant k during the forward travel in the trellis.
- Backward state metrics $\beta_k(s_k)$ which correspond to the probability of the trellis to be in state s_k at the instant k during the backward travel in the trellis.
- LLRs which represent the log likelihood ratios of the information bits and the coded bits or symbols.

The forward and backward state metrics also denoted as recursion metrics are recursively calculated at each state s_k and at each instant k as follow:

$$\begin{aligned} \alpha_k(s_k) &= \sum_{s_{k-1}} \alpha_{k-1}(s_{k-1}) \gamma_k(s_{k-1}, s_k), \quad k = 0, ..., K_b - 1 \\ \beta_k(s_k) &= \sum_{s_{k+1}} \beta_{k+1}(s_{k+1}) \gamma_k(s_k, s_{k+1}), \quad k = K_b - 1, ..., 0 \end{aligned} \tag{1.20}$$

where K_b corresponds to the block length. Their initialization depends on the initial state and final state in the trellis. Uniform initialization is used in case of unknown state.

$$\alpha_0(s) = \begin{cases} 1 & \text{if } s = s_0 \\ 0 & \text{if } s \neq s_0 \end{cases} \tag{1.21}$$

$$\beta_{K_b}(s) = \begin{cases} 1 & \text{if } s = s_0 \\ 0 & \text{if } s \neq s_0. \end{cases} \tag{1.22}$$

The branch metric $\gamma(s_k, s_{k+1})$ for a corresponding symbol or bit $u_k = j$ is given by:

$$\gamma(s_k, s_{k+1}) = p(y_k/u_k) Pr^a(u_k = j, s_k, s_{k+1}). \tag{1.23}$$

If case of no transition between the two states s_k and s_{k+1}, the branch metric $\gamma(s_k, s_{k+1})$ is equal to zero. $p(y_k/u_k)$ represents the channel probability transition. In case of Gaussian channel, this probability is expressed by:

$$p(y_k/u_k) = \prod_{i=1}^{n} \left(\frac{1}{\sigma\sqrt{2\pi}} \exp - \frac{(y_{k,i} - x_{k,i})^2}{2\sigma^2} \right) = K \exp \frac{\sum_{i=1}^{n} y_{k,i} x_{k,i}}{\sigma^2}. \tag{1.24}$$

All the metrics can be expressed in the logarithmic domain by taking the logarithm of these values [5]. Let $\bar{\gamma}$, $\bar{\alpha}$ and $\bar{\beta}$ the logarithms of γ, α and β, respectively. Using

the Jacobian logarithm: $\log\left(e^a + e^b\right) = \max^*(a,b) = \max(a,b) + \log\left(1 + e^{-|a-b|}\right) = \max(a,b) + f_c(|a-b|)$, these logarithmic metrics can be computed as [5]:

$$\begin{aligned}
\bar{\gamma}_k(s_k, s_{k+1}) &= \log \gamma_k = K' + \sum_{i=1}^{n} y_{k,i} x_{k,i} + L_k^a(u_k = j), \\
\bar{\alpha}_k(s_k) &= \log \alpha_k = \overset{*}{\underset{(s_{k-1})}{\max}} \left(\bar{\alpha}_{k-1}(s_{k-1}) + \bar{\gamma}_k(s_{k-1}, s_k)\right), \quad k = 0...K_b - 1 \\
\bar{\beta}_k(s_k) &= \log \beta_k = \overset{*}{\underset{(s_{k+1})}{\max}} \left(\bar{\beta}_{k+1}(s_{k+1}) + \bar{\gamma}_k(s_k, s_{k+1})\right), \quad k = K_b - 1...0 \\
L_k(j) &= \overset{*}{\underset{d_k=j}{\max}} \left(\bar{\alpha}_{k-1}(s_{k-1}) + \bar{\gamma}_k(s_{k-1}, s_k) + \bar{\beta}_k(s_k)\right) \\
&\quad - \overset{*}{\underset{(s_{k-1}, s_k)}{\max}} \left(\bar{\alpha}_{k-1}(s_{k-1},) + \bar{\gamma}_k(s_{k-1}, s_k) + \bar{\beta}_k(s_k)\right).
\end{aligned} \tag{1.25}$$

The recursion metrics are then initialized to zeros in the initial state and to infinity on others states. This algorithm is referred to as log-MAP algorithm. We note that the correction factor $f_c(|a-b|) = \log\left(1 + e^{-|a-b|}\right)$ can be approximated by a precomputed look-up table (LUT). When using the max-log approximation: $\log\left(e^a + e^b\right) \approx \max(a,b)$, the algorithm is denoted as max-log-MAP algorithm.

For a binary turbo code, the decoder computes the *a posteriori* log likelihood ratio (LLR) of each information bit u_k that represents the probability of the bit to be zero over the probability to be one.

$$\begin{aligned}
L(u_k) &= \overset{*}{\underset{u_k=0}{\max}} \left(\bar{\alpha}_{k-1}(s_{k-1}) + \bar{\gamma}_k(s_{k-1}, s_k) + \bar{\beta}_k(s_k)\right) \\
&\quad - \overset{*}{\underset{u_k=1}{\max}} \left(\bar{\alpha}_{k-1}(s_{k-1}) + \bar{\gamma}_k(s_{k-1}, s_k) + \bar{\beta}_k(s_k)\right).
\end{aligned} \tag{1.26}$$

The component decoder exchanges only the extrinsic LLR $L_e(u_k)$ given by:

$$L_e(u_k) = L(u_k) - L_a(u_k) - L_{sys}(u_k), \tag{1.27}$$

where $L_a(u_k)$ and $L_{sys}(u_k)$ correspond respectively to the *a priori* information from the other decoder and the systematic information.
MAP algorithm requires a large amount of memory since the recursive metrics have to be stored for an entire block. In addition, it requires a long decoding latency. Therefore, sliding windows approach has been proposed, in which the recursion metric is computed within a window size with an estimated initialization value [53].

1.3.1.3 QPP interleaver

The interleaver has a critical impact on the performance of the turbo decoder. The main role of the interleaver is to control the minimum Hamming distance, and to reduce the degree of correlation between the soft output of the component encoders. The reduction of the degree of correlation increases the probability for error correction and the performance of turbo code.

The interleavers can be classified into two categories: random interleavers and deterministic interleavers. Random interleavers permute the information bits in a pseudo random manner given near Shannon performance. However, the design of random interleaver is not suitable for hardware implementations specially with large block length.

Deterministic interleavers are proposed to avoid this problem by using an analytical expressions to perform the permutation of information bits. The simplest examples of deterministic interleavers are block interleavers and linear interleavers.

Quadratic permutation polynomial (QPP) interleaver has been proposed as a contention free interleaver suitable for parallel decoding of turbo codes. Therefore, the QPP-interleaver resolves the memory contention problem when multiple processes try to read or write into the same memory address. It allows a degree of parallel processing equal to every factor of the interleaver length.
The QPP interleaver is based on permutation polynomials over integer rings. For a given block length K_b, the x^{th} interleaved position can be expressed as:

$$f(x) = \left(f_1 x + f_2 x^2\right) \mod K_b, \tag{1.28}$$

where f_1 and f_2 are non-negative integers and depend on the block length K_b ($f_1, f_2 < K_b$). f_1 and f_2 are always an odd and even number respectively. In LTE standard, there are 188 block sizes ranging from 40 to 6144 [48]. All block sizes K_b are even integer divisible by 4 and 8. The block length is also divisible by 16, 32 and 64 when $K_b \geq 512$.

The contention free property $f(x)$ of the interleaver or de-interleaver for a block length K_b divided into N_{sb} sub-blocks of length L_{sb} ($K_b = N_{\text{sb}} L_{\text{sb}}$) satisfies the following constraint:

$$\left\lfloor \frac{f(x + iL_{\text{sb}})}{L_{\text{sb}}} \right\rfloor \neq \left\lfloor \frac{f(x + jL_{\text{sb}})}{L_{\text{sb}}} \right\rfloor, \tag{1.29}$$

where $\lfloor x \rfloor$ denotes the nearest integer equal or less than the variable x, $0 \leq x < L_{\text{sb}}$, $0 \leq i, j < N_{\text{sb}}$, and $i \neq j$. This inequality indicates that for a fixed offset x, exactly one cell is accessed from each processor as shown in Figure 1.7.

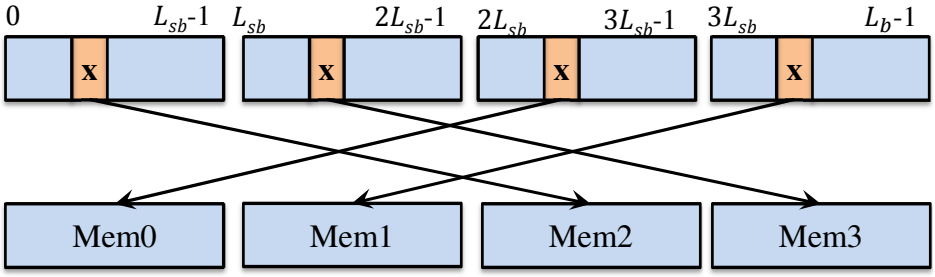


Figure 1.7: QPP interleaver propriety.

1.3.1.4 Rate matching

The initial rate of LTE turbo encoder is $R_c = 1/3$. In order to get other code rates, rate matching method is used to perform puncturing or repetition [48]. The rate matching method consists first in permuting the systematic and parity bits using dedicated sub-block interleavers. Then, the interleaved systematic bits are written into a circular buffer followed by the interleaved and interlaced parity bits. A bit selection and pruning step is realized by the circular buffer starting from a certain point specified by redundancy version (rv) to get the desired code rate as shown in Figure 1.8. If the end of the circular buffer is reached and more bits are needed, the selection continues by wrapping around at the beginning of the buffer.

Each sub-block interleaver is based on row-column permutations, and consists of 32 columns. The bits of each stream are written row-by-row into a matrix with 32 columns and $\lceil (K_b + 4)/32 \rceil$ rows (dummy bits are padded if the matrix is not full). After column permutation, the bits are read out from the matrix column by column.

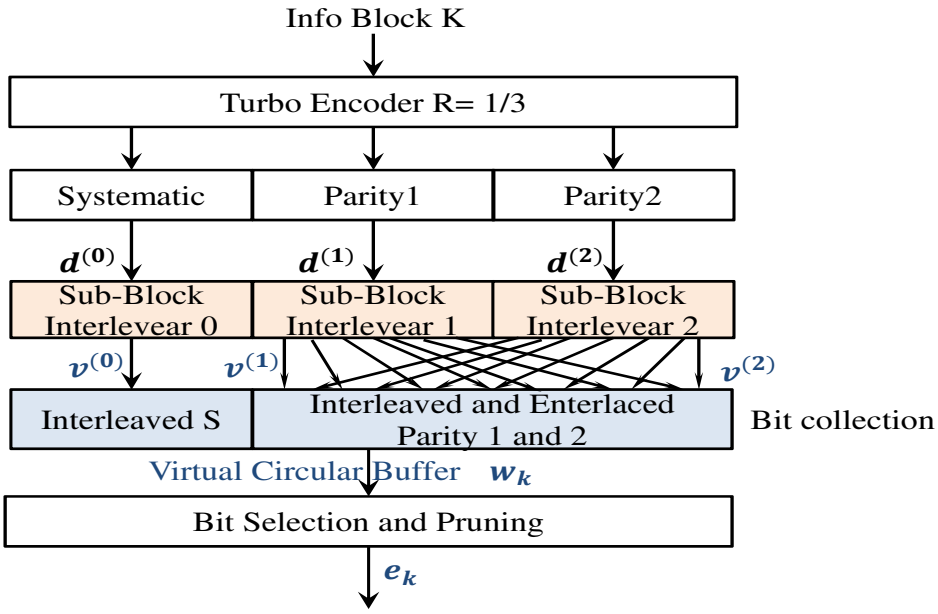


Figure 1.8: Rate matching process in LTE.

1.3.2 LDPC codes

LDPC codes are a class of linear error correcting block codes first proposed by Gallager in 1963 [3]. Their main advantage is that they provide performance which is very close to the capacity. Furthermore they are suited for parallel implementations. Therefore, LDPC decoders were rediscovered due to their low decoding complexity thanks to iterative process applied to turbo decoding[47].

1.3.2.1 LDPC encoder

LDPC codes are defined by a sparse parity check matrix $\mathbf{H}_\mathrm{p}$ that contains only a very small number of non-zero entries satisfying the equation:

$$\mathbf{H}_\mathrm{p}\mathbf{c} = 0. \tag{1.30}$$

There are two different representations of LDPC codes: matrix representation and graphical representation. The graphical representation is introduced by Tanner [54]. Tanner graphs are bipartite graphs containing two types of nodes: check nodes and variable nodes. It consists of M_b check nodes (CN) which correspond to the number of parity bits and N_b variable nodes (VN) corresponding to the number of bits in a codeword. Check node i is connected to variable node j if the element h_{ij}^{LDPC} of $\mathbf{H}_\mathrm{p}$ is equal to 1. Figure 1.9 gives an example of Tanner graph and its corresponding parity check matrix.

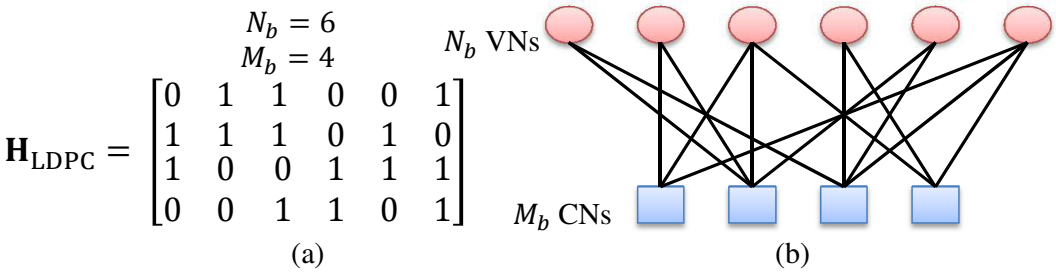


Figure 1.9: LDPC representation, **(a)** Matrix representation, **(b)** Tanner graph.

LDPC encoder generates codeword $\mathbf{c}$ by simply multiplying the data stream $\mathbf{u}$ with a generator matrix $\mathbf{G}_{\text{LDPC}}$:

$$\mathbf{c} = \mathbf{u}\mathbf{G}_{\text{LDPC}}. \tag{1.31}$$

The generator matrix and the parity check matrix are orthogonal and satisfy:

$$\mathbf{G}_{\text{LDPC}}\mathbf{H}_{\text{p}}^{T} = 0. \tag{1.32}$$

Different methods can be used to generate the parity check matrix which can be either regular or irregular. An LDPC code is said to be regular if the check node and bit node degrees are constant and irregular if they are not. The node degree corresponds to the number of ones in the rows (for check nodes) or columns (for variable nodes) of the parity check matrix.
In general, the generator matrix can be found by performing Gauss-Jordan elimination on $\mathbf{H}_{\text{p}}$ to obtain it in the form:

$$\mathbf{H}_{\text{p}} = \left[\mathbf{A} \quad \mathbf{I}_{N_b - K_b}\right], \tag{1.33}$$

$$\mathbf{G}_{\text{LDPC}} = \left[\mathbf{I}_{K_b} \quad \mathbf{A}^{T}\right]. \tag{1.34}$$

The LDPC code can be also encoded using the parity check matrix directly by transforming it into upper triangular form and using back substitution [55].

1.3.2.2 LDPC decoder

Optimal maximum *a posteriori* decoding of LDPC codes is practically an infeasible problem. As an alternate, LDPC decoding is iteratively carried out by passing messages between the nodes of the Tanner graph, known as message passing algorithms. The algorithm is denoted as bit-flipping when the messages are binary, and as belief propagation when the messages are probabilities. The belief propagation is referred to as sum-product decoding since probabilities are represented in form of LLRs which allow the calculation of messages using sum and product operations. Min-sum algorithm (MSA) is proposed as a simplification of sum-product algorithm in order to reduce the implementation complexity of the decoder. An overview of various belief propagation decoding algorithms and their reduced complexity derivatives are presented in [6].

LDPC decoder can be viewed as two concatenated decoders: variable node (VN) decoder and check node (CN) decoder as illustrated in Figure 1.10.

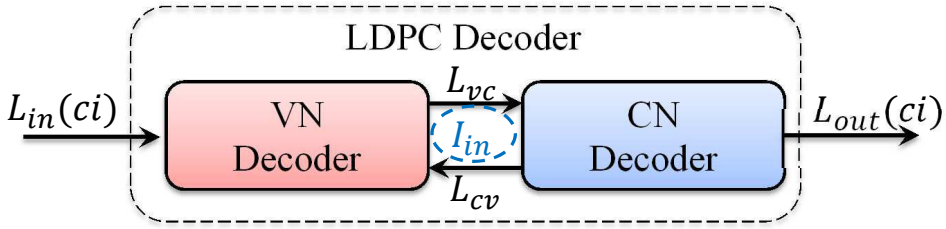


Figure 1.10: Block diagram of LDPC decoder.

Sum-product algorithm (SPA)

The aim of SPA is to compute the maximum *a posteriori* probability for each codeword bit by iteratively computing and exchanging extrinsic information between the variable and check nodes as shown in Figure 1.11.

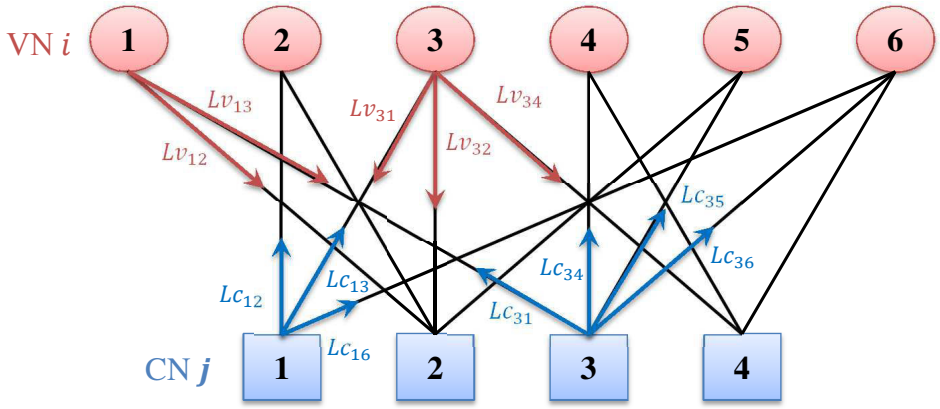


Figure 1.11: Iterative decoding between VNs and CNs.

Let $L_{v_{ij}}$ the variable node message from variable node i to check node j and $L_{c_{ji}}$ the check node message from check node j to variable node i. At the first iteration, the input to the LDPC decoder is the log-likelihood ratio $L(c_i)$ of the codeword $\mathbf{c}$ which is used as an initial value of the extrinsic variable node message, $L_{v_{ij}} = L(c_i)$. At the k^{th} iteration, the algorithm can be summarized as follows:

1. Each check node j uses the message received from its neighboring and computes the extrinsic message to the variable node i:

$$L_{c_{ji}} = 2\tanh^{-1}\left(\prod_{i\prime\in V_j\setminus i}\tanh\left(\frac{L_{v_{i\prime j}}}{2}\right)\right). \tag{1.35}$$

2. Each variable node i updates their extrinsic information which is then sent back to the check node j in the next iteration:

$$L_{v_{ij}} = L(c_i) + \sum_{j\prime\in C_i\setminus j} L_{c_{j\prime i}}. \tag{1.36}$$

3. The *a posteriori* LLR of each codeword bit is computed as:

$$L_{\text{p}}(c_i) = L(c_i) + \sum_{j\prime\in C_i} L_{c_{j\prime i}}, \tag{1.37}$$

where V_j and C_i denote the set of adjacent variable nodes connected to the check node j, and the set of adjacent check nodes connected to the variable node i, respectively. The decoding algorithm alternates between check node processing and variable node processing until a maximum number of iterations has been reached or until the parity check condition is satisfied and outputs the *a posteriori* LLR of the codeword L_{p}.

In order to reduce the computational complexity, the SPA can be simplified using the min-sum algorithm (MSA), where the updated messages from check nodes to variable nodes $L_{cv_{ji}}$ are computed as follows [56]:

$$L_{c_{ji}} \approx \left(\prod_{i\prime \in V_j \backslash i} \operatorname{sign}\left(\mathrm{L}_{\mathrm{v_{i\prime j}}}\right) \right) \min_{i\prime} \left| L_{v_{i\prime j}} \right|. \tag{1.38}$$

1.4 Orthogonal frequency division multiplexing

Orthogonal frequency division multiplexing is an effective multi-carrier technique [7] to deal with the frequency selectivity in multipath fading environments with an efficient use of spectrum. OFDM technique was rapidly used in wireless communication standards due to its simple implementation through the use of discrete Fourier transform (DFT).

OFDM modulation consists in transmitting the modulated symbols over multiple orthogonal sub-carriers. First the symbols are demultiplexed through a serial to parallel converter into N parallel sub-streams. An inverse fast Fourier transform (IFFT) is then applied, which results in a set of N time domain complex symbols. A cyclic prefix (CP) of length Δ is added before transmission through the frequency selective channel. The cyclic prefix must be chosen larger than the delay spread of the channel to completely remove the ISI.

At the receiver, the reverse operations are performed to demodulate the OFDM symbols. The CP is removed and the FFT is applied to convert the signal back to the frequency domain. Therefore, the transmitted signal over a frequency selective channel is converted into a transmission over N flat-fading channels. Figure 1.12 illustrates the basic principle of OFDM modulation and demodulation.

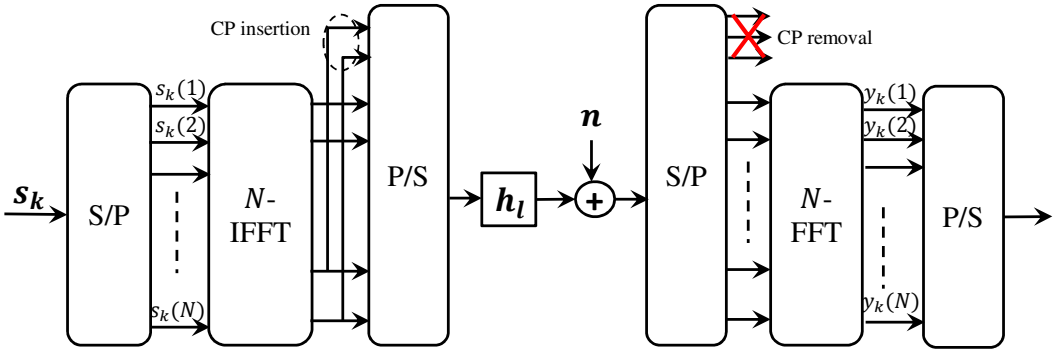


Figure 1.12: Block diagram of OFDM modulation and demodulation.

Assuming that the channel is linear time invariant, the received signal at the k^{th}

OFDM symbol can be expressed by:

$$\begin{bmatrix} y_k(1) \\ \vdots \\ \vdots \\ \vdots \\ y_k(N+\Delta) \end{bmatrix} = \begin{bmatrix} h_{L-1} & \cdots & h_0 & 0 & & \\ 0 & \ddots & & \ddots & \ddots & \\ & \ddots & \ddots & & \ddots & 0 \\ & & 0 & h_{L-1} & \cdots & h_0 \end{bmatrix} \begin{bmatrix} s_{k-1}(N+\Delta-L+1) \\ \vdots \\ s_{k-1}(N+\Delta) \\ s_k(1) \\ \vdots \\ s_k(N+\Delta) \end{bmatrix} + \begin{bmatrix} n_k(1) \\ \vdots \\ \vdots \\ \vdots \\ n_k(N+\Delta) \end{bmatrix}, \tag{1.39}$$

where $[s_{k-1}(N+\Delta-L+1), \quad ..., \quad s_{k-1}(N+\Delta)]$ corresponds to the interference from the $(k-1)^{\text{th}}$ OFDM symbol. The insertion of a cyclic prefix larger than the delay spread of the channel allows to avoid this interference and to get a circular channel matrix. The received signal can be therefore expressed as:

$$\begin{bmatrix} y_k(1) \\ \vdots \\ \vdots \\ \vdots \\ y_k(N) \end{bmatrix} = \begin{bmatrix} h_0 & 0 & \cdots & 0 & h_{L-1} & \cdots & h_1 \\ h_1 & \ddots & \ddots & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & h_{L-1} \\ h_{L-1} & & h_1 & h_0 & 0 & & 0 \\ 0 & \ddots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{L-1} & \cdots & h_1 & h_0 \end{bmatrix} \begin{bmatrix} s_k(1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ s_k(N) \end{bmatrix} + \begin{bmatrix} n_k(1) \\ \vdots \\ \vdots \\ \vdots \\ n_k(N) \end{bmatrix}. \tag{1.40}$$

Through the use of the unitary DFT matrix, the circular convolution in time domain is transformed into a multiplication in the frequency domain. Therefore, the equalization processes is reduced to simple one tap equalization per each sub-carrier.

$$\mathbf{y}_k = \mathbf{H}\mathbf{s}_k + \mathbf{n}_k. \tag{1.41}$$

In practical systems, the number of modulated sub-carriers is less than the FFT/IFFT size, null sub-carriers are therefore added at the border of the spectrum in order to avoid the overlap between adjacent channels. We denote by N_c the number of useful sub-carriers and by N the FFT/IFFT size.

OFDM technique is sensitive to Doppler shifting, which leads to loss of orthogonality between sub-carriers (Inter-carrier interference). Synchronization techniques must be applied in this case to handle the carrier frequency offset (CFO) and symbol time offset. Through this work, we assume perfect time and frequency offset synchronization. The high peak-to-average power ratio (PAPR) is another drawback of OFDM technique. It is due to constructive and destructive combinations of independent sub-carriers and leads to increase the non-linear distortions, which imposes to work with enough power back-off to avoid distortion.

1.5 Multiple-input multiple-output techniques

MIMO systems use multiple antennas at the transmitter and at the receiver in order to achieve high data rate transmission or high reliability. The capacity of MIMO system increases linearly with the minimum number between the transmit and the

received antennas. In addition, MIMO systems exploit the space diversity improving the robustness of the system and increasing the quality of transmission.

MIMO techniques can be divided into different categories which depends on the availability of the channel state information (CSI) at the transmitter and at the receiver. In this section, we review the different MIMO transmission schemes, the capacity and the achieved gain.

1.5.1 MIMO transmission schemes

By assuming the CSI availability at the receiver, MIMO schemes can be classified into two main categories: diversity techniques and spatial multiplexing (SM) techniques. In the diversity techniques, the same data is redundantly transmitted over more than one transmit antennas or received by more than one antenna, thereby improving the transmission reliability. The spatial multiplexing, on the other hand, separates the data into several streams which are then simultaneously transmitted into the channel, thereby increasing data rate transmission. To combine the advantages of both transmission modes, the precoding techniques can be used [57].

Space time coding

Space time codes are used to generate the signal redundancy in order to exploit the full diversity of MIMO channels and assure high link reliability. Two main families of space time coding exist: space-time trellis code (STTC) and space-time block code (STBC).

Space-time trellis code [58] is a generalization of trellis coded modulation for MIMO system. It provides both coding gain and diversity gain by combining the channel coding and space time coding. However, STTC presents an exponential complexity with large trellis dimension which makes it less attractive for practical implementation.

STBC uses a form of repetition coding to decouple the non-orthogonal channels into a set of orthogonal SIMO channels. This scheme presents the advantage of exploiting the transmit diversity of the system with the use of low complexity receiver based on maximum ratio combining (MRC) algorithm.
For each STBC, Q_s symbols are transmitted over T interval time leading to a code rate $R_s = \frac{Q_s}{T}$. The channel is in general assumed to be constant over the duration T in order to guarantee the orthogonality of the code. The STBC is said to be full rate, when the rate is equal to one ($R_s = 1$). The STBC can exploit a maximum diversity equal to $N_t \times N_r$.

The first orthogonal STBC (OSTBC) was proposed by Alamouti for two transmit antennas and one receive antenna [59]. The OSTBC provides maximum spatial diversity equal to $N_t \times N_r$ and the decoding is reduced to a simple linear decoding. OSTBC is generally represented by a matrix $\mathbf{X}$ of dimension $N_t \times T$ satisfying:

$$\mathbf{X}\mathbf{X}^H = c\left(|s_1|^2 + |s_2|^2 + ... + |s_k|^2\right)\mathbf{I}_{N_t}. \tag{1.42}$$

The matrix $\mathbf{X}$ of the Alamouti code is given by:

$$\mathbf{X}_{\text{Al}} = \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix}. \tag{1.43}$$

Tarokh extended OSTBC proposed by Alamouti to support different number of transmit antennas [60]. These schemes achieve full spatial diversity with a space-time coding rate less than one ($R_s < 1$).

The main drawbacks of STBC is the limitation of the code rate ($R_s \leq 1$). In order to increase the rate, the principle of orthogonality must be little ignored. Therefore, quasi-orthogonal space time codes have been proposed [61, 62]. However, quasi-orthogonal STBC introduces a little co-antenna interference (CAI) which necessitates a more complex receiver to achieve optimal performance.
Space time block coding may be equivalently used in frequency domain denoted as space frequency block coding (SFBC) as used in LTE.

Spacial multiplexing

The objective of SM schemes is to achieve higher spectral efficiency by transmitting multiple data streams in parallel over the antennas. The first SM scheme was proposed in [11]. This scheme is denoted as Diagonal-BLAST (D-BLAST) where the signal information is separated into N_t streams having the same spectral efficiency. Each stream is then encoded, independently modulated and cyclically distributed in time over the N_t transmit antennas. The decoder of a D-BLAST is based on the principle of interference cancellation. This decoding layer is complex, which is why Vertical-BLAST (V-BLAST) [13] system was developed. In V-BLAST system, the data stream is encoded, interleaved and demultiplexed into N_t substreams. The allocation of a substream to an antenna is fixed during the entire transmission. V-BLAST system can be considered as an extension in space of bit-interleaved coded modulation (BICM) [63] and it is referred to as space-time bit-interleaved coded modulation (ST-BICM).

Linear dispersion codes

Linear dispersion code (LDC) has been proposed as a trade-off between diversity and multiplexing techniques [64]. LDC maximizes the mutual information taking into consideration the spatial diversity of MIMO channel. Its basic idea is to linearly superpose space time symbols. In addition, their linear structure simplify the decoding of V-BLAST system. LDC distributes in time and space a vector of Q_s complex symbols $\mathbf{s} = [s_1 \quad s_2 \quad ... \quad s_{Q_s}]^T$ belonging to a linear constellation (QAM, PSK). The space time coding matrix $\mathbf{X}$ is given by:

$$\mathbf{X} = \sum_{q=1}^{Q_s} \left(\mathrm{Re}\,(s_q)\, \mathbf{A}_q + \sqrt{(-1)}\, \mathrm{Im}\,(s_q)\, \mathbf{B}_q \right), \tag{1.44}$$

where $\mathrm{Re}\,(.)$ and $\mathrm{Im}\,(.)$ represent the real and the imaginary parts of the variables, respectively. $\mathbf{A}_q$ and $\mathbf{B}_q$ are $N_t \times T$ linear dispersion (generator) matrices.

The family of linear dispersion codes is very large and comprises many existing space-time codes. It can be considered as a generalization of space time codes. For example, the spatial multiplexing scheme is actually a linear dispersion code, where $T = 1$ and $Q_s = N_t$.

When the CSI is available at the transmitter and at the receiver, singular value decomposition (SVD) can be used in order to transform the MIMO channel onto a set

of parallel sub-channels. The data streams can be therefore independently demodulated at the receiver side. This scheme is commonly known as beam-forming. Water-filling technique can be also used at the transmitter in order to maximize the data rate subject to the power constraint [65].

There is also the family of MIMO techniques without CSI at the transmitter and at the receiver. These techniques are based on differential coding [66, 67]. However, the complexity of the decoding makes them less attractive. For the rest of this work, only MIMO techniques with CSI at the receiver are considered.

1.5.2 MIMO gains

The spatial diversity of MIMO system may be exploited to improve link reliability and spectral efficiency. The gains enabled by MIMO technologies can be categorized as follows [57, 1].

Diversity gain

The diversity gain is directly related to the fading channels. There are three main types of diversity that can be exploited :i) temporal diversity, which is caused by the delay spread of the signal, ii) frequency diversity, which is caused by the Doppler spread, and iii) spatial diversity exploited by sending the signal over different paths in the space. The exploitation of MIMO diversity increases the reliability of communication system by transmitting or receiving multiple copies of the same signal over independently fading links. The diversity gain of the system can be defined as [57]:

$$G_{\mathrm{d}} = -\lim_{\mathrm{SNR}\longrightarrow\infty} \frac{\log_2 p_e\left(\mathrm{SNR}\right)}{\log_2\left(\mathrm{SNR}\right)}, \tag{1.45}$$

where p_e is the probability of error. The maximum diversity gain is equal to the number of independent links in MIMO system which is equal to the product of the number of transmit and receive antennas, $G_{\mathrm{d}}^{\max} = N_r N_t$.

Multiplexing gain

The spatial multiplexing aims to increase the capacity of the communication system by sending independent symbols simultaneously from different transmit antennas. Spatial multiplexing gain is defined by:

$$G_{\mathrm{SM}} = \lim_{\mathrm{SNR}\longrightarrow\infty} \frac{R\left(\mathrm{SNR}\right)}{\log_2\left(\mathrm{SNR}\right)}, \tag{1.46}$$

where R is the transmission rate of the system. The SM gain is upper bounded by the minimum number of transmit and receive antennas, $G_{\mathrm{SM}}^{\max} = \min(N_t, N_r)$. This bound corresponds to the rank of the channel or equivalently to the number of parallel sub-channels.

In [57], it was shown that it is impossible to maximize G_{SM} and G_{d} at the same time. The optimal tradeoff is given by:

$$G_{\mathrm{d}} = (N_t - G_{\mathrm{SM}})(N_r - G_{\mathrm{SM}}). \tag{1.47}$$

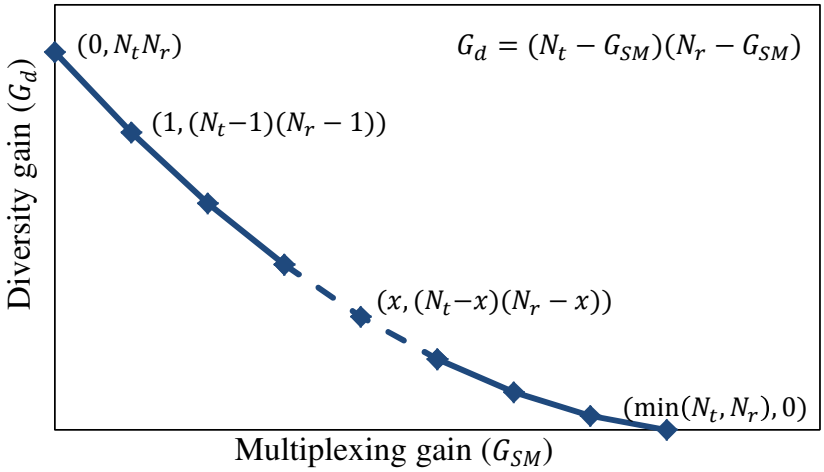


Figure 1.13: Diversity gain versus multiplexing gain.

Figure 1.13 illustrates the diversity gain versus the multiplexing gain. We can see that an increase of the diversity gain will inevitably lead to a smaller multiplexing gain.

Graphically, the diversity gain is traduced by the slope of bit error rate curve, this slop increases when the diversity gain increases. Whereas, coding gain is traduced by horizontal shifting of the bit error rate curve with respect to SNR.

Array gain

The array gain also referred to as power gain depends on the number of transmit and receive antennas. It results in an increase in the average receive SNR by combining all signals using multiple receive antennas. Therefore, the average SNR grows proportionally to the number of receive antennas. In addition, the array gain may be exploited at the transmitter when CSI is available through the use of precoding techniques.

1.5.3 MIMO channel capacity

The channel capacity is defined as the maximum rate for which information can be transmitted reliably over the channel. Its concept is first introduced by Shannon in 1948 [45], who showed that there is a theoretical limit of channel capacity. This limit corresponds to the maximum of the mutual information $I(\mathbf{s}, \mathbf{y})$ between the transmit and the receive signals $\mathbf{s}$ and $\mathbf{y}$:

$$C = \max_{p_s(\mathbf{s})} I(\mathbf{s}, \mathbf{y}) \tag{1.48}$$

where p_s is the probability density function of the transmit signals.

In [65], Telatar gives theoretical expressions for MIMO channel capacity in Rayleigh fading channels. He shows that this capacity linearly increases with the minimum number of transmit or receive antennas. This capacity largely depends on the channel model. Hence, the instantaneous capacity is considered when the channel $\mathbf{H}$ is deterministic. If the channel is a random process, the ergodic capacity is defined. Meanwhile, the outage capacity is considered when the channel is constant over a large time interval.

Different schemes are also possible: when CSI is known at the receiver and at the transmitter (closed loop); when the CSI is available at the receiver only (open loop); or when there is no information at the transmitter neither at the receiver.

Singular value decomposition (SVD) of the channel

The channel matrix $\mathbf{H}$ has a singular value decomposition:

$$\mathbf{H} = \mathbf{U}\Sigma\mathbf{V}^H, \tag{1.49}$$

where $\mathbf{U}$ and $\mathbf{V}$ are $N_r \times N_r$ and $N_t \times N_t$ unitary matrices respectively, Σ is $N_r \times N_t$ diagonal matrix with real and positive values. The diagonal elements (σ_i) of Σ are the singular values of the matrix $\mathbf{H}$.
Using the SVD decomposition, the MIMO system with N_t transmit antennas and N_r receive antennas can be seen as an equivalent system consisting of min (N_t, N_r) independent parallel sub-channels in which the channel response has a power gain equal to the eigenvalues $\lambda_i = \sigma_i^2$. The min (N_t, N_r) is known as the rank of the channel r_h.

Deterministic channel capacity

By assuming that the channel matrix $\mathbf{H}$ is perfectly known at the receiver, the mutual information between $\mathbf{s}$ and $\mathbf{y}$ can be written as [65]:

$$I(\mathbf{s}, \mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y}|\mathbf{s}) = \log_2 \det\left(I_{N_r} + \frac{\sigma_s^2}{\sigma_n^2}\mathbf{H}\mathbf{R}_{ss}\mathbf{H}^H\right). \tag{1.50}$$

The channel capacity is defined by the maximum of $I(\mathbf{s}, \mathbf{y})$ over the possible distributions $p(\mathbf{s})$ subject to the transmit power constraints, where $\mathbf{R}_{ss}$ is the auto-correlation matrix of $\mathbf{s}$. The transmit power is normalized, $tr(\mathbf{R}_{ss}) = 1$. Then, the instantaneous capacity of deterministic MIMO channel is expressed as:

$$C_{\text{det}}(\mathbf{H}) = \log_2\left(\det\left(\mathbf{I} + \frac{\sigma_s^2}{\sigma_n^2}\mathbf{H}\mathbf{R}_{ss}\mathbf{H}^H\right)\right). \tag{1.51}$$

In practice, the channel state information has to be learned by the receiver. This can be done using pilot signals to estimate all $N_t N_r$ sub-channels which lower the potential gains of MIMO systems.

In the case of perfect CSI at the transmitter, MIMO system can be then transformed into a set of parallel SISO channels using SVD decomposition. The transmit signal is pre-processed with $\mathbf{V}$ in the transmitter and the received signal is post-processed with $\mathbf{U}^H$ in the receiver. The MIMO channel capacity with CSI at the transmitter is given by:

$$C_{\text{det}}^{\text{CL}}(\mathbf{H}) = \sum_{i=1}^{r_h} \log_2\left(1 + \lambda_i \frac{\tilde{\mathbf{R}}_{ss}}{\sigma_n^2}\right), \tag{1.52}$$

where $\tilde{\mathbf{R}}_{ss} = \mathbf{V}^H\mathbf{R}_{ss}\mathbf{V}$. The capacity can be maximized using water-filling technique in order to determine the diagonal of $\tilde{\mathbf{R}}_{ss}$ subject to the transmit power constraints.

For system with no CSI at the transmitter, $\mathbf{R}_{ss} = \frac{1}{N_t}\mathbf{I}_{N_t}$, the channel capacity can be written as [65]:

$$C_{\text{det}}^{\text{OL}}(\mathbf{H}) = \log_2 \det\left(\mathbf{I}_{N_r} + \frac{\sigma_s^2}{N_t\sigma_n^2}\mathbf{H}\mathbf{H}^H\right) = \sum_{i=1}^{r_h} \log_2\left(1 + \lambda_i \frac{\sigma_s^2}{N_t\sigma_n^2}\right). \tag{1.53}$$

Ergodic and outage channel capacity

In practical applications, the channel matrix is not fixed but randomly changes. Consequently, the channel capacity will depend not only on the SNR but also on the channel realizations. For fixed number of independent channel realizations N_s, outage capacity is considered. Such a capacity represents the maximum rate at which communication is possible with a probability of error less than p_e:

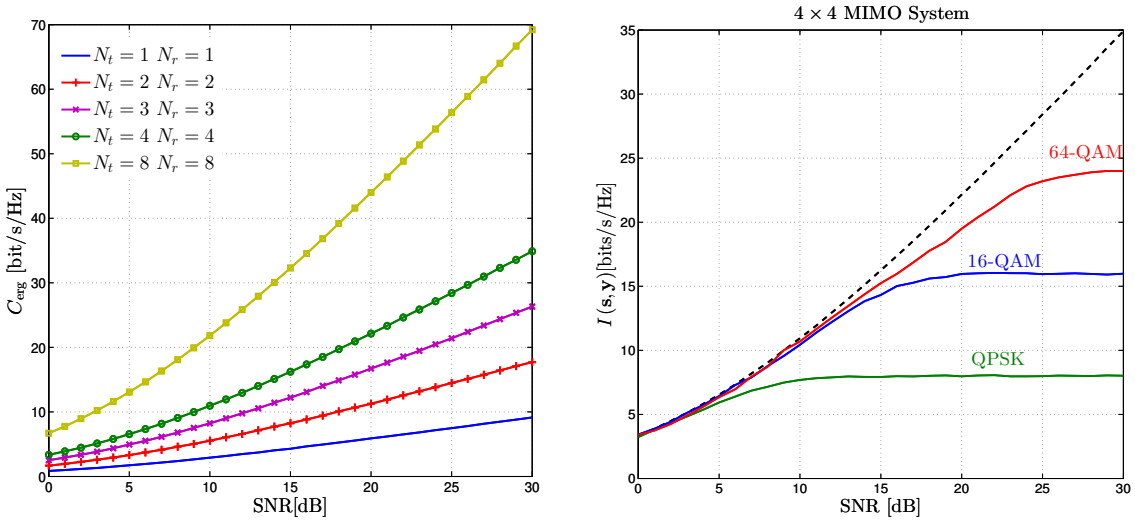
$$C_{\text{out}}\left(N_s, p_e\right) = \left\{ Pr\left(\frac{1}{N_s} \sum_{n=1}^{N_s} C_{\text{det}}\left(\mathbf{H}_n\right) < R \right) \leq p_e \right\}. \tag{1.54}$$

When the transmission interval is long enough to observe the full channel realizations, we speak about ergodic channel capacity given by:

$$C_{\text{erg}} = \lim_{N_s \to \text{inf}} C_{\text{out}}\left(N_s, 0\right) = \mathbb{E}\left\{C_{\text{det}}\left(H\right)\right\} = \mathbb{E}\left\{ \log_2\left(\det\left(\mathbf{I} + \frac{\sigma_s^2}{\sigma_n^2} \mathbf{H} \mathbf{R}_{ss} \mathbf{H}^H \right) \right) \right\}. \tag{1.55}$$

Figure 1.14a shows the ergodic channel capacity of MIMO systems ($N_r = N_t$) with 1, 2, 3, 4 and 8 antennas. We show that such a capacity linearly increases with the number of antennas at high SNR.

In case of the coded MIMO system with QAM constellation, the channel capacity will be upper bounded by the transmission rate $R = R_c Q N_t$. Therefore, the mutual information I is used to measure the data rate achieved by different constellations and coding schemes [27]. The mutual information I versus SNR for different constellations is plotted in Figure 1.14b for 4×4 MIMO system. For a given data rate, the modulation order and coding scheme must be chosen in order to obtain a mutual information very close to the capacity curves.



(a) **(b)**

Figure 1.14: Ergodic channel capacity of MIMO system with different number of transmit and receive antennas **(a)** and average Mutual information according to the SNR for 4x4 **(b)** MIMO system using 3 constellations.

1.6 MIMO system model and detection problematic

1.6.1 MIMO system model

We consider a MIMO system based on bit-interleaved coded modulation (BICM) scheme [68] with N_t transmit antennas and N_r receive antennas ($N_r \geq N_t$) as depicted in Figure 1.15.

At the transmitter, the data stream $\mathbf{u}$ is first encoded and punctured with a coding rate R_c into a sequence of $\mathbf{c}$ coded bits. The channel encoder can be a convolutional encoder, a turbo encoder or an LDPC encoder. Then, the encoded stream is randomly interleaved and mapped into complex symbols of a $M_c = 2^Q$ quadrature amplitude modulation (QAM) constellation, where Q is the number of bits per symbol. The resulting sequence of symbols is mapped into N_t dimensional symbol vectors $\mathbf{s} \in 2^{Q \cdot N_t}$ using either space-time block coding (STBC) schemes or spatial multiplexing (SM) schemes. Each transmit symbol vector $\mathbf{s} = [s_1, s_2, ..., s_{N_t}]^T$ is then associated with $N_t Q$ bits $x_{i,b} \in \{0, 1\}$, $i = 1...N_t$, $b = 1...Q$, where the indexes b and i correspond to the b^{th} bit of the i^{th} symbol of $\mathbf{s}$.

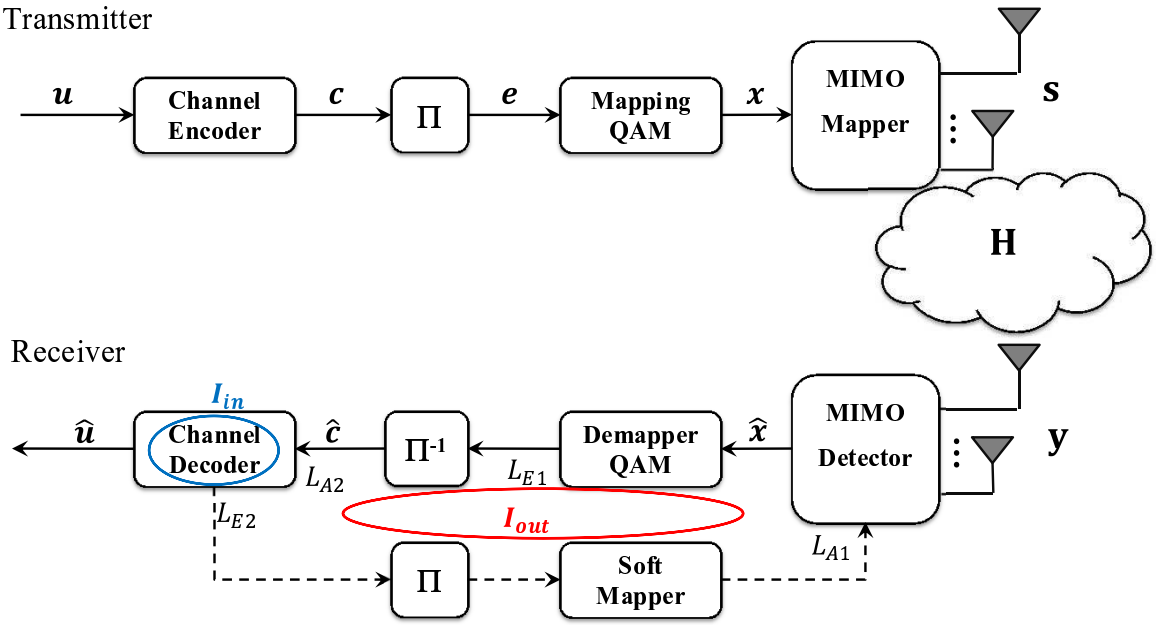


Figure 1.15: MIMO system block diagram using bit-interleaved coded modulation with iterative detection and decoding.

Herein, the SM-based MIMO system is considered without loss of generality. It is assumed that the channels experience independent Rayleigh fading, and the transmitter does not require any channel state information (CSI). The transmit power is normalized so that $\mathbb{E}\left\{\mathbf{s}\mathbf{s}^H\right\} = E_s/N_t \mathbf{I}_{N_t}$. The transmission information rate is $R_c \cdot N_t \cdot Q$ bits per channel use. The received vector $\mathbf{y} = [y_1, y_2, ..., y_{N_r}]^T$ can be represented by:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1.56}$$

where $\mathbf{n} = [n_1, n_2, ..., n_{N_r}]^T$ is an independent and identically distributed (i.i.d.) additive white Gaussian noise (AWGN) vector with zero mean and σ_n^2 variance ($N_0 =$

σ_n^2).

$$n \sim \mathcal{CN}(0, \sigma_n^2). \tag{1.57}$$

$\mathbf{H}$ is an $N_r \times N_t$ channel matrix, assumed to be perfectly known at the receiver, with independent elements h_{ij} of zero mean and unit variance complex Gaussian random variables $h \sim \mathcal{CN}(0, 1)$. h_{ij} represents the channel response between the j^{th} transmit antenna and i^{th} receive antenna as represented in Figure 1.16.

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} \dots & h_{N_t 1} \\ h_{21} & h_{22} \dots & h_{N_t 2} \\ \vdots & & \\ h_{N_t 1} & h_{N_t 2} \dots & h_{N_t N_r} \end{bmatrix}. \tag{1.58}$$

Gray labeling is used for mapping the bits to constellation symbols. In Gray labeling,

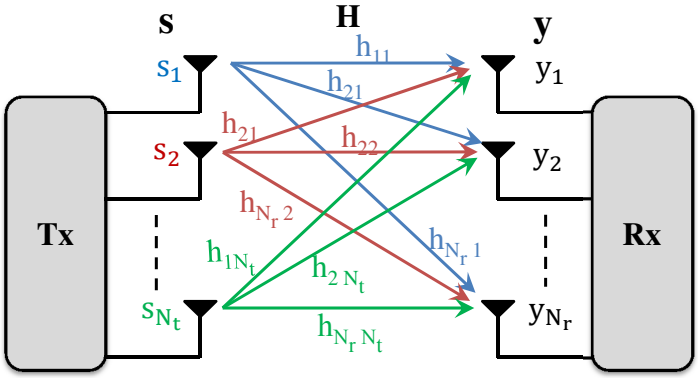


Figure 1.16: MIMO channel.

two neighbor constellation points differ in a single bit. This feature ensures a maximum average mutual information at the output of the detector if no *a priori* information from the decoder is available. Obviously, the mapping bits to constellation symbols can be independently done for real and imaginary parts. This feature will be useful to transform the system model into an equivalent real system model as follows:

$$\begin{bmatrix} \mathrm{Re}\,(\mathbf{y}) \\ \mathrm{Im}\,(\mathbf{y}) \end{bmatrix} = \begin{bmatrix} \mathrm{Re}\,(\mathbf{H}) & -\mathrm{Im}\,(\mathbf{H}) \\ \mathrm{Im}\,(\mathbf{H}) & \mathrm{Re}\,(\mathbf{H}) \end{bmatrix} \begin{bmatrix} \mathrm{Re}\,(\mathbf{s}) \\ \mathrm{Im}\,(\mathbf{s}) \end{bmatrix} + \begin{bmatrix} \mathrm{Re}\,(\mathbf{n}) \\ \mathrm{Im}\,(\mathbf{n}) \end{bmatrix}, \tag{1.59}$$

where $\mathrm{Re}\,(.)$ and $\mathrm{Im}\,(.)$ represent the real and the imaginary parts of the variables, respectively. In this equivalent real system model, the QAM constellation can be viewed as two PAM constellations, and the matrix dimension is hence doubled. In [69], the real model was revealed to be more efficient for the implementation of the sphere decoder and it will be used for the system model in the case of sphere decoder.

At the receiver, reverse operations are carried out to recover the transmitted information. The MIMO detector takes the received signal $\mathbf{y}$ to estimate the transmit vector. The symbols are then demodulated and decoded. An iterative process can be performed to improve the performance of the system which will be the topic of next chapters. The MIMO detector produces either hard-decision of transmitted symbols or computes soft information. In case of hard estimates, the detector is referred to as hard-decision MIMO detector discussed in chapter 2, while the detector that computes

soft information is referred to as soft-input soft-output detector and will be investigated in chapter 3.

In case of frequency selective channel, multipath propagation leads to ISI, the received signal can be expressed as:

$$y_j(t) = \sum_{l=1}^{L} \sum_{i=1}^{N_t} h_{ji}(l) s_i)(t-l) + n_j(t), \tag{1.60}$$

where $y_j(t)$ is the received signal by the j^{th} antennas, h_{ji} denoted the l^{th} path between the j^{th} received antenna and i^{th} transmit antenna and L is the number of taps. For low complexity equalization of the frequency selective channel, OFDM technique is combined with MIMO technique in our work. Hence, flat fading channel is assumed over each sub-carriers, thus limiting the complexity of the receiver caused by ISI in fading channels. The received signal in the case of MIMO-OFDM system can be then expressed as:

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{s}_k + \mathbf{n}_k \quad k = 1...N_c, \tag{1.61}$$

where k is the index of sub-carriers. For simplicity, the sub-carrier index k is omitted in the sequel. The bit signal to noise ratio $\frac{E_b}{N_0}$ is given by:

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} \frac{N_r}{N_t} \frac{1}{QR_c}. \tag{1.62}$$

Spatial correlation

Spatial correlation refers to the correlation of the propagation channels between different antennas. The capacity of the system depends on the spatial correlation that greatly depends on the space between antennas. This spatial correlation must be taken into consideration to calculate the spatially correlated channel matrix $\mathbf{R_H}$ defined for a flat fading channel by:

$$\mathbf{R_H} = \mathbb{E}_{\mathbf{H}} \left\{ \mathrm{Vec}(\mathbf{H}) \mathrm{Vec}(\mathbf{H})^{\mathrm{H}} \right\}, \tag{1.63}$$

where Vec($\mathbf{H}$) is the vectorization matrix obtained by superposing the columns of $\mathbf{H}$. The size of $\mathbf{R_H}$ increases with the dimension of MIMO system.
For simplification, Kronecker model is used where the correlations between transmit antennas and receive antennas are assumed independent and separable. Using this assumption, $\mathbf{R_H}$ becomes the Kronecker product between transmit and receive correlation matrices $\mathbf{R_s}$, $\mathbf{R_y}$:

$$\mathbf{R_H} = \mathbf{R_s} \otimes \mathbf{R_y} = \mathbf{R_y}^{1/2} \mathbf{H} (\mathbf{R_s}^{1/2})^T. \tag{1.64}$$

1.6.2 Detection problematic

At the receiver, a linear superposition of separately transmitted symbols is observed. The task of the receiver is therefore to recover the transmitted vector $\mathbf{s}$ from the noisy received vector $\mathbf{y}$, which corresponds to the general detection problematic.

From a geometrical point of view, the detection problem can be seen as a lattice search problem which consists in finding the lattice point **Hs** that minimizes the Euclidean distance between this point and the received vector **y** [16]. For a generated matrix **H** , the complex lattice for any transmitted vector **s** is defined by :

$$\mathcal{L}(\mathbf{H}) = \left\{\mathbf{z} = \mathbf{Hs} | \mathbf{s} \in 2^{QN_t}\right\} = \left\{s_1\mathbf{H}_{:,1} + s_2\mathbf{H}_{:,2} + ... + s_{N_t}\mathbf{H}_{:,N_t} | s_n \in 2^Q\right\} \quad (1.65)$$

where $\mathbf{H}_{:,i}$ are the columns of **H** that denote the basis vectors of the lattice [70], with N_t and N_r the rank and the dimension of the lattice. Figure 1.17 illustrates an example of two dimensional real lattices with orthogonal basis and correlated basis. In the case of well conditioned channels, the boundaries of decision regions form an orthogonal grid. This allows the signal to be easily estimated. However, in the case of a noisy channels, the decision regions are generally polytopes which require more comparison to get the estimated signals.

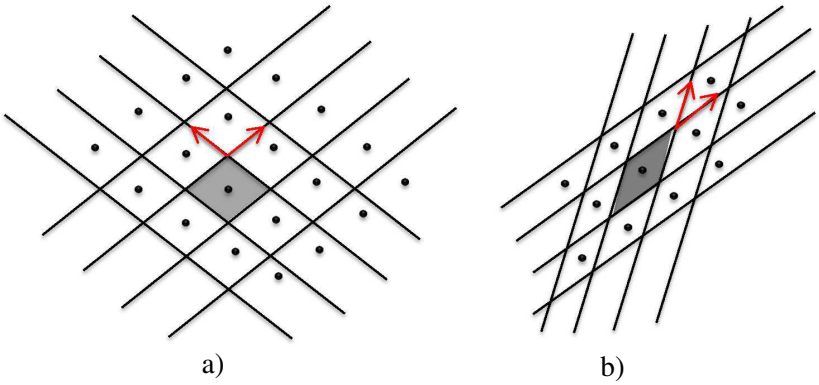


Figure 1.17: Examples of 2-dimensional real lattices: **(a)** orthogonal bases and **(b)** correlated bases.

1.7 Conclusion

In this chapter, we have presented some background knowledge which is required for the following chapters of this thesis. Wireless communication systems have been first described followed by introducing channel models as well as some channel characteristics. We have then introduced channel coding which will be extensively used in our further studies. The basic concept of OFDM techniques has been also presented. Furthermore, we have given an overview of MIMO schemes which are nowadays combined with channel coding and OFDM techniques to achieve better performance and higher throughput. The general MIMO system model and the detection problematic are finally described which will be carried out in details in the following chapters.

Chapter 2

Hard-Decision MIMO Detection

Contents

MIMO systems have been considered as an effective solution for reliable high data rate transmissions in future wireless communication systems. The use of multiple antennas is able to provide high data rate and to enhance the transmission quality at no cost in transmit power and frequency spectrum.

However, as presented in the previous chapter, a linear superposition of the transmitted signals is observed at the receiver. The role of the receiver is to detect the original signal from the distorted received one with minimum errors while approaching the channel capacity.

Therefore, it is interesting for us to carry out investigations on different MIMO detection algorithms proposed in the literature, which is the aim of this chapter. Among the optimal maximum likelihood (ML) detection, we consider the relevant sub-optimal families of MIMO detection: linear detection, interference cancellation detection and tree-search based detection. Their associated advantages, limitations and performance are compared and discussed.

2.1 Maximum likelihood detection

The maximum likelihood (ML) detector is the optimum hard-decision MIMO detector [9]. It uses an exhaustive search to find the transmitted vector $\mathbf{s}$ from the received one among 2^{QN_t} possible symbol combinations.

ML detector selects $\mathbf{s}$ such that the *a posteriori* probability $p(\mathbf{s}/\mathbf{y})$ is maximized:

$$\hat{\mathbf{s}}_{\mathrm{ML}} = \arg \max_{\mathbf{s} \in 2^{QN_t}} p(\mathbf{s}/\mathbf{y}) = \arg \max_{\mathbf{s} \in 2^{QN_t}} \left\{ \frac{p(\mathbf{y}/\mathbf{s}) p(\mathbf{s})}{p(\mathbf{y})} \right\}. \tag{2.1}$$

By assuming a white Gaussian noise, the conditioned probability density function $p(\mathbf{y}/\mathbf{s})$ of $\mathbf{y}$ given $\mathbf{s}$ can be expressed as:

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{(\pi N_0)^{N_r}} \exp\left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right). \tag{2.2}$$

By further assuming that the symbol vectors are equiprobable, the optimal ML solution in equation (2.1) is reduced to:

$$\hat{\mathbf{s}}_{\mathrm{ML}} = \arg \min_{\mathbf{s} \in 2^{QN_t}} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2. \tag{2.3}$$

This ML solution corresponds to the closest point search problem by finding the smallest Euclidean distance between the received symbol vector $\mathbf{y}$ and each possible transmitted vectors.

ML detector offers optimal performance on uncoded MIMO system. It has been shown that ML detector can only be feasible with low-order modulations and small number of antennas [71, 72]. However, its complexity exponentially increases with respect to the number of transmit antennas and constellation sizes. For example, in the case of a 2×2 MIMO system with 4-QAM, $2^{2\times 2} = 16$ possible solutions need to be compared. In the case of a 4×4 MIMO system with 16-QAM, there are $2^{4\times 4} = 65,536$ possible solutions.

A number of MIMO detectors have been therefore proposed with reduced complexity as will be discussed in the following sections.

2.2 Linear detection

The linear detector (LD), referred to as equalizer, simply uses a filtering matrix $\mathbf{G}_{\mathrm{LD}}$ to invert the effect of MIMO channel matrix and multiply it by the received vector [10]. The equalized symbol $\tilde{\mathbf{s}}$ is then quantized into the nearest constellation symbol $\hat{\mathbf{s}}$ as depicted in Figure 2.1. The detection of each symbol is independently done where the detection problem is decomposed into N_t single-antenna detection problems, which leads to significant performance loss. The filtering matrix can be constructed using zero-forcing (ZF) or minimum-mean square error (MMSE) criteria.

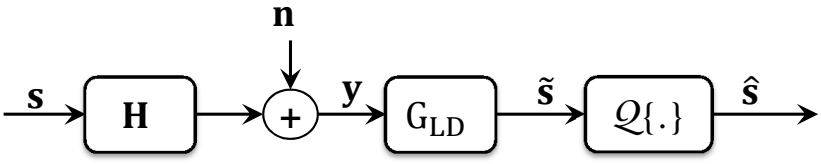


Figure 2.1: Block diagram of linear detector.

2.2.1 ZF equalizer

ZF equalizer multiplies the received vector by the pseudo inverse of the channel matrix, resulting in a full cancellation of the interference between transmitted symbols. The ZF filtering matrix $\mathbf{G}_{\mathrm{ZF}}$[11] is given by:

$$\mathbf{G}_{\mathrm{ZF}} = \mathbf{H}^{\dagger} = (\mathbf{H}^{H}\mathbf{H})^{-1}\mathbf{H}^{H}. \tag{2.4}$$

ZF detector presents a low computational complexity. However, it may introduce noise enhancement when the channel is ill-conditioned, which leads to significant performance degradations.

2.2.2 MMSE equalizer

MMSE equalizer was introduced to alleviate the colored noise induced by ZF equalizer. The filtering matrix $\mathbf{G}_{\mathrm{MMSE}}$ is therefore constructed to fulfill MMSE criterion given by [12]:

$$\mathbf{G}_{\mathrm{MMSE}} = \arg\min_{\mathbf{G}} \mathbb{E}\left\{\|\mathbf{G}\mathbf{y} - \mathbf{s}\|^{2}\right\}. \tag{2.5}$$

Using the principle of orthogonality between the received vector and the noise vector: $\mathbb{E}\left\{(\mathbf{G}_{\mathrm{MMSE}}\mathbf{y} - \mathbf{s})\mathbf{y}^{H}\right\} = 0$, $\mathbf{G}_{\mathrm{MMSE}}$ can be computed as:

$$\mathbf{G}_{\mathrm{MMSE}} = \left(\mathbf{H}^{H}\mathbf{H} + \frac{\sigma_n^2}{\sigma_s^2}\mathbf{I}_{N_t}\right)^{-1}\mathbf{H}^{H}, \tag{2.6}$$

where σ_s^2 and σ_n^2 are the variances of the transmitted vector and noise vector respectively. The equalized symbols $\tilde{s}_k$ at the output of the equalizer are generally associated with a bias factor β_k in addition to some residual noise plus interferences η_k:

$$\tilde{s}_k = \beta_k s_k + \eta_k. \tag{2.7}$$

MMSE equalizer balances the residual interference and the noise enhancement and improves the BER performance compared to ZF equalizer. However, at high SNR, MMSE solution theoretically converges to ZF solution.

Although linear detectors are attractive in terms of computational complexity, they are not able to exploit the diversity order of ML detector due to the independent detection of symbols. The use of previous detected symbols to cancel the interference terms in the received signal can improve the performance of the system. This approach is denoted as interference cancellation detection or also referred to as decision-feedback detection as discussed in the next section.

2.3 Interference cancellation detection

The main principle of interference cancellation detection is to recursively detect the transmitted symbols in order to refine the mitigation of interference. The detection layer corresponds to a transmit antenna. The detection takes place layer by layer with the interference of previously detected layers removed from the received signal before detecting the next layer. Two categories of interference cancellation have been proposed. The first approach, denoted as V-BLAST algorithm, uses a set of linear filters as originally proposed [11, 13]. The other approach is based on the QR decomposition (QRD) of the channel matrix $\mathbf{H}$ [14, 15].

2.3.1 V-BLAST: Successive interference cancellation

In V-BLAST algorithm [11, 13], a successive cancellation step and interference nulling step are used to detect the transmitted symbols. The algorithm starts by detecting the symbol from the first layer using ZF or MMSE filtering matrix and subtracting it from the received vector. This procedure is repeated until the detection of all symbols.

This method suffers from error propagation issue, where uncorrected detected symbol in first layer will create errors in the following layers. Therefore, superior performance can be obtained by ordering the detection of symbols. Such an approach is denoted as ordered SIC (OSIC) [13] or V-BLAST detector. First, the symbol with the highest signal to interference noise ratio which suffers from least noise amplification is detected. This symbol corresponds to the row of the filtering matrix $\mathbf{G}$ having the least Euclidean norm. The OSIC algorithm can be summarized as follows:

Set $\mathbf{y}^{(1)} = \mathbf{y}$, for $i = 1$ to N_t do

1. $\mathbf{G} = \mathbf{G}_{\text{ZF}}$ or $\mathbf{G} = \mathbf{G}_{\text{MMSE}}$
2. $k_i = \arg\min_{j, j \neq k_i} \|\mathbf{G}_{j,:}\|^2$
3. $\tilde{s}_{k_i} = \mathbf{G}_{k_i,:}\mathbf{y}^{(i)}$
4. $\hat{s}_{k_i} = \mathcal{Q}\{\tilde{s}_{k_i}\}$
5. $\mathbf{y}^{(i+1)} = \mathbf{y}^{(i)} - \mathbf{H}_{:,k_i}\hat{s}_{k_i}$
6. $\mathbf{H}_{:,k_i} = 0$, $\mathbf{H} = [..., \quad \mathbf{H}_{:,k_i-1}, \quad \mathbf{H}_{:,k_i+1}, \quad ...]$,

where $\mathcal{Q}$ is the quantization operation to the nearest constellation symbol. At each iteration i, the interference due to the k_i^{th} detected symbol is canceled out from the received signal. Then, its corresponding column $\mathbf{H}_{:,k_i}$ in the channel matrix $\mathbf{H}$ is removed. This strategy is repeated up to find all components of $\mathbf{s}$ as illustrated in Figure 2.2.

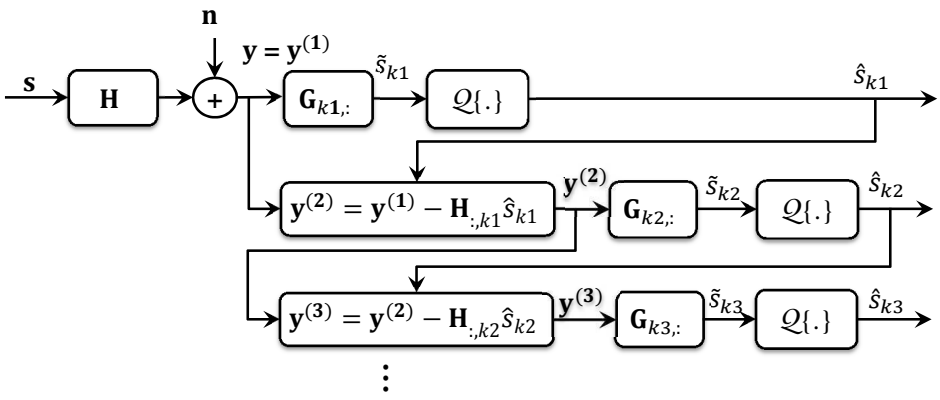


Figure 2.2: Block diagram of V-BLAST algorithm.

The computational complexity of V-BLAST detection algorithm is $\mathcal{O}(N_t^4)$ since it requires multiple calculations of the pseudo-inverse of the channel matrix. Therefore several techniques have been proposed to reduce its complexity to $\mathcal{O}(N_t^3)$ [73, 74].

2.3.2 QR decomposition-based detection

QRD-based detection is another algorithm of successive interference cancellation. The channel matrix $\mathbf{H}$ can be decomposed into two matrices $\mathbf{Q}$ and $\mathbf{R}$, where $\mathbf{Q}$ is an $(N_r \times N_t)$ unitary matrix ($\mathbf{Q}^H\mathbf{Q} = \mathbf{I}_{N_t}$) and $\mathbf{R}$ is an $(N_t \times N_t)$ upper triangular matrix with real-positive entries on its diagonal. The main idea of QRD-based detection is to transfer the MIMO system from a non-causal system to causal one, and thus splitting the computational complexity into preprocessing step and detection step.

At the preprocessing step, the QRD is first performed ($\mathbf{H} = \mathbf{QR}$). Then, the received symbol vector $\mathbf{y}$ is multiplied by $\mathbf{Q}^H$ prior to the detection step:

$$\mathbf{Q}^H\mathbf{y} = \mathbf{Q}^H(\mathbf{Hs} + \mathbf{n}) = \mathbf{Rs} + \mathbf{Q}^H\mathbf{n}, \tag{2.8}$$

$$\tilde{\mathbf{y}} = \mathbf{Rs} + \tilde{\mathbf{n}}, \tag{2.9}$$

where $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$ and $\tilde{\mathbf{n}} = \mathbf{Q}^H\mathbf{n}$. We note that the covariance of the noise term remains unchanged since $\mathbf{Q}$ is an unitary matrix.

Since $\mathbf{R}$ is an upper triangular matrix, the transmit symbol vector $\mathbf{s}$ is estimated from the vector $\tilde{\mathbf{y}}$ and the matrix $\mathbf{R}$ using Gauss elimination algorithm [75]. The modified received signal at the i^{th} layer can be written as:

$$\tilde{y}_i = \mathbf{R}_{i,i}\tilde{s}_i + \sum_{i=j+1}^{N_t} \mathbf{R}_{i,j}\hat{s}_j. \tag{2.10}$$

Therefore, the SIC estimated symbol at the i^{th} layer is given by:

$$\hat{s}_i^{\text{SIC}} = \mathcal{Q}\left\{\frac{1}{\mathbf{R}_{i,i}}\left(\tilde{y}_i - \sum_{j=i+1}^{N_t} \mathbf{R}_{i,j}\hat{s}_j^{\text{SIC}}\right)\right\}, \tag{2.11}$$

where $\mathcal{Q}\{.\}$ denotes the quantization operation to the nearest constellation symbol. By assuming correct previous detected symbols, the interference can be perfectly canceled in each layer. Figure 2.3 illustrates the principle of QRD-based detection.

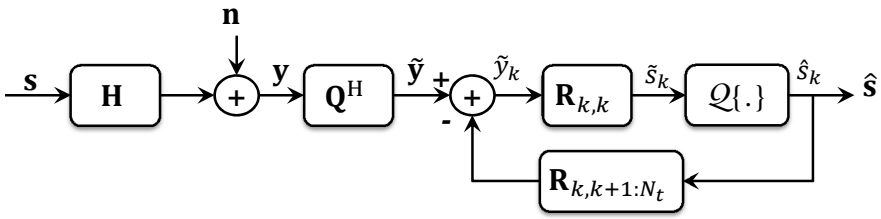


Figure 2.3: Block diagram of QRD-based detector.

Various QRD algorithms have been proposed in the literature with variable implementation advantages, namely: Gram-Schmidt (GS), Householder (HH) and Givens rotations (GR) [75, 76]. The GS decomposition consists of two steps, orthogonalization and normalization, in which the matrix $\mathbf{Q}$ is obtained before the matrix $\mathbf{R}$. The HH triangularization is based on a reflexion matrix. The matrix $\mathbf{H}$ is iteratively treated in order to get $\mathbf{R}$, then $\mathbf{Q}$ if it is required. The GR is usually used since it is well suitable for parallel implementation. This technique is based on the cancellation of the elements of $\mathbf{H}$ to get the triangular matrix $\mathbf{R}$ [76].

Sorted QR decomposition

Layer ordering and regularization can be used with QRD [14, 15]. A well suitable ordering would reduce the error propagation issue and consequently improve the performance of the system. The optimal detection order which maximizes the SNR in each step (maximizes $\mathbf{R}_{k,k}$ for $k = N_t, ..., 1$) can be found by performing $\mathcal{O}(N_t^2/2)$ QRDs of permutations of $\mathbf{H}$ [77]. By assuming that all the previous decisions are correct, the SNR in each step can be expressed by:

$$\mathrm{SNR_k} = \frac{\mathbf{R}_{\mathrm{k,k}}}{\sigma_{\mathrm{n}}^2}. \tag{2.12}$$

In [14], an efficient heuristic approach is proposed that comes close to the error performance of the global optimization. The corresponding detector is denoted as sorted QRD (SQRD) detector in the sequel. This SQRD algorithm is basically an extension of the modified Gram-Schmidt procedure by reordering the columns of the channel matrix prior to each orthogonalization step. The basic idea is to find the permutation of $\mathbf{H}$ that minimizes the diagonal element $\mathbf{R}_{k,k}$ with k running from 1 to N_t, leaving all $\mathbf{R}_{j,j}$ with $j < k$ unchanged. SQRD algorithm minimizes the diagonal elements in every decomposition step and thereby intends maximal diagonal elements $\mathbf{R}_{i,i}$ in the succeeding steps $i > k$. It can be summarized as follows:

Set $\mathbf{R} = \mathbf{0}$, $\mathbf{Q} = \mathbf{H}$, $\mathbf{P} = [1, ..., N_t]$

1. for $i = 1$ to N_t do
2. $k_i = \arg\min_{j=i:N_t} \left\|\mathbf{Q}_{:,j}\right\|^2$
3. Exchange columns i and k_i in $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{P}$
4. $\mathbf{R}_{i,i} = \sqrt{\left\|\mathbf{Q}_{:,j}\right\|^2}$
5. $\mathbf{Q}_{:,i} = \mathbf{Q}_{:,i}/\mathbf{R}_{i,i}$
6. for $j = i + 1$ to N_t do
7. $\mathbf{R}_{i,j} = \mathbf{Q}_{:,i}^H \mathbf{Q}_{:,j}$
8. $\mathbf{Q}_{:,j} = \mathbf{Q}_{:,j} - \mathbf{R}_{i,j}\mathbf{Q}_{:,i}$.

The efficiency of layer ordering can be further improved using MMSE criterion. With the use of extended system model [73]:

$$\mathbf{H}_e = \begin{bmatrix} \mathbf{H} \\ \sigma_n \mathbf{I}_{N_t} \end{bmatrix} \quad \mathbf{y}_{\mathrm{e}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_{N_t} \end{bmatrix}, \tag{2.13}$$

the classical MMSE solution will be equal to ZF solution of the extended model:

$$\mathbf{G}_{e,\mathrm{ZF}}\mathbf{y}_e = (\mathbf{H}_e^H\mathbf{H}_e)^{-1}\mathbf{H}_e^H\mathbf{y}_e = (\mathbf{H}^H\mathbf{H} + \sigma_n^2\mathbf{I}_{N_t})^{-1}\mathbf{H}^H\mathbf{y} = \mathbf{G}_{\mathrm{MMSE}}\mathbf{y}. \tag{2.14}$$

Therefore, SQRD is applied to the extended channel matrix [15]:

$$\mathbf{H}_e\mathbf{P} = \mathbf{Q}_e\mathbf{R}_e = \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \mathbf{R}_e, \tag{2.15}$$

where $\mathbf{P}$ is an $N_t \times N_t$ permutation matrix, $\mathbf{Q}_e$ is an $(N_r + N_t) \times N_t$ unitary matrix with orthogonal columns, and $\mathbf{R}_e$ is an $N_t \times N_t$ upper-triangular matrix. The matrix $\mathbf{Q}$

can be decomposed into two matrix: $\mathbf{Q}_1$ (non-orthogonal matrix) of dimension $N_r \times N_t$ and $\mathbf{Q}_2$ (upper triangular matrix) of dimension $N_t \times N_t$ [15].

By multiplying the received vector by $\mathbf{Q}_e^H$, we get:

$$\mathbf{Q}_e^H \mathbf{y}_e = \mathbf{Q}_1^H \mathbf{y} = \mathbf{R}_e \tilde{\mathbf{s}} + \tilde{\mathbf{n}}, \tag{2.16}$$

where $\tilde{\mathbf{s}} = \mathbf{P}^{-1}\mathbf{s}$,and $\tilde{\mathbf{n}} = -\sigma_n \mathbf{Q}_2^H \mathbf{s} + \mathbf{Q}_1 \mathbf{n}$ is the effective noise vector. Note that $\mathbf{Q}$ is unitary, but $\mathbf{Q}_1$ and $\mathbf{Q}_2$ will not be unitary. Therefore, the effective noise vector $\tilde{\mathbf{n}}$ is no longer i.i.d. circularly symmetric complex Gaussian distributed with variance σ_n^2. The lower triangular matrix $\mathbf{Q}_2^H$ constitutes the remaining interference that can not be removed by the successive interference cancellation procedure.

The optimal detection sequence must be now chosen such that the signal to interference and noise (SINR) ratio is maximized, which leads to a minimal estimation error for the corresponding detection step. The MMSE-SQRD offers a low complexity solution, but it does not necessary lead to the optimal detection order. Subsequently, the post-sorting algorithm (PSA) has been introduced in order to achieve the optimal ordering with an additional computational complexity [15].

2.4 Tree-search based detection

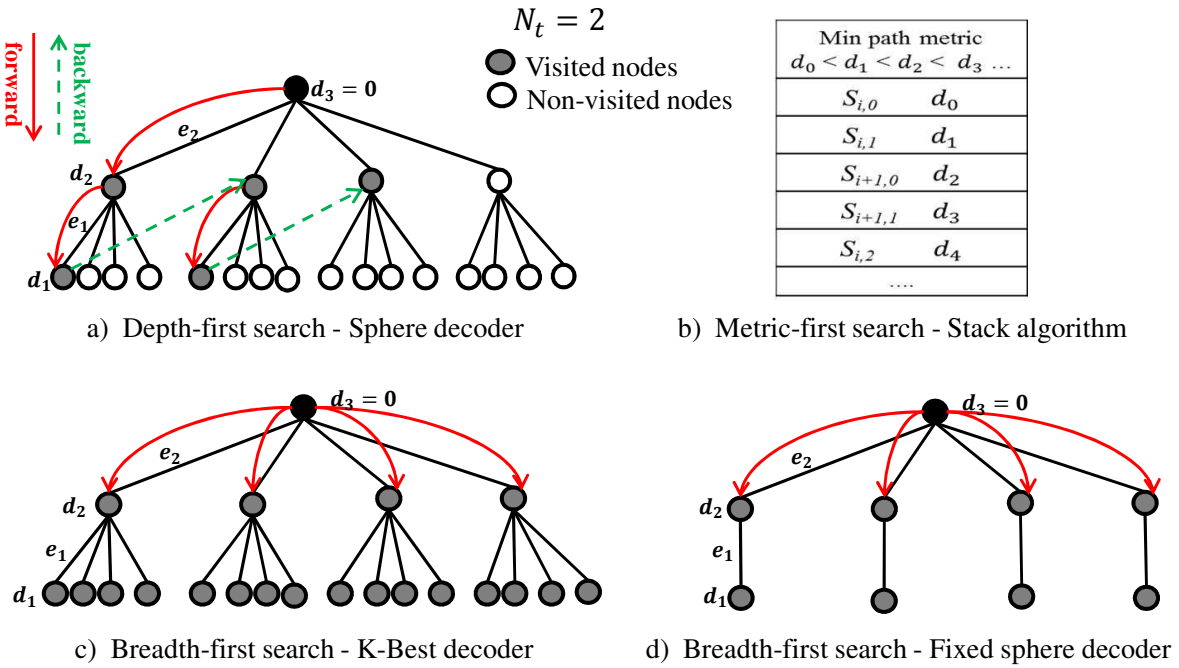
The detection problem can be transformed into a tree-search problem [17, 20, 78, 16, 79]. Several tree-search based detection algorithms are proposed in the literature to achieve near ML performance with low computational complexity. These algorithms can be classified into 3 categories: depth-first search, metric-first search and breadth-first search approach (Figure 2.4). The classical sphere decoding is a depth-first approach, the stack algorithm is a metric-first search, while the K-Best decoding and fixed sphere decoding are commonly seen as breadth-first approaches.

2.4.1 Depth-first search - Sphere decoder

The sphere decoder (SD) is an efficient depth-first search method to solve the detection problem and achieve near optimal performance with polynomial average computational complexity for a large range of SNR [80].
The SD was originally developed by Pohst in 1981 [81] for the computation of minimal length lattice vectors. In 1985, improved methods for calculating the short lattice vectors were introduced by Fincke and Pohst [17]. It was then used for ML estimation in [82, 83]. Viterbo and Biglieri applied the Fincke-Pohst (FP) algorithm to lattice decoding in 1993 [84]. Schnorr and Euchner [20] proposed a refinement to the FP algorithm in 1994. In 1999, Viterbo and Boutros used lattice code decoding in fading channels [78], and in 2000, Damen et al. used lattice code decoder for space-time codes.

The fundamental idea of SD is to limit the search space of ML solution to an hypersphere of radius r_s around the received vector as illustrated in Figure 2.5. Therefore, only lattice points that lie inside the hypersphere are tested instead of testing all the hypotheses of the transmitted signal, reducing the computational complexity:

$$\hat{\mathbf{s}}_{\mathrm{SD}} = \arg \min_{\mathbf{s} \in 2^{QN_t}} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \leq r_{\mathrm{s}}^2 \right\}. \tag{2.17}$$



a) Depth-first search - Sphere decoder

b) Metric-first search - Stack algorithm

c) Breadth-first search - K-Best decoder

d) Breadth-first search - Fixed sphere decoder

Figure 2.4: Tree-search strategies.

The channel matrix $\mathbf{H}$, as we have previously seen, can be decomposed into two matrix $\mathbf{Q}$ and $\mathbf{R}$, $\mathbf{H} = \mathbf{QR}$. Therefore by using the QRD, the detection problem is equivalent to:

$$\hat{\mathbf{s}}_{\mathrm{SD}} = \arg \min_{\mathbf{s} \in 2^{QN_t}} \left\{ \|\tilde{\mathbf{y}} - \mathbf{Rs}\|^2 \leq r_s^2 \right\}. \tag{2.18}$$

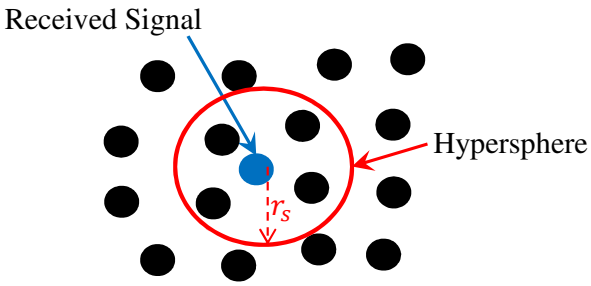


Figure 2.5: Sphere decoder principle.

Exploiting the triangular nature of $\mathbf{R}$, the Euclidean distance metric in equation (2.18), $d_1 = \|\tilde{\mathbf{y}} - \mathbf{Rs}\|^2$, can be recursively evaluated through the accumulated partial Euclidean distance (PED) d_i with $d_{N_t+1} = 0$ as follows:

$$d_i = d_{i+1} + \left| \tilde{y}_i - \sum_{j=i}^{N_t} \mathbf{R}_{i,j} s_j \right|^2 = d_{i+1} + |e_i|^2. \tag{2.19}$$

This process can be illustrated by a tree with $N_t + 1$ levels as depicted in Figure 2.6, where level i corresponds to the i^{th} transmit antenna. The tree-search starts at the root level with the first child node at level N_t corresponding to the symbol transmitted

by the N_t^{th} antenna. The partial Euclidean distance d_{N_t} in equation (2.19) is then computed. If d_{N_t} respects the sphere radius constraint r_s, the search continues at level $N_t - 1$ and steps down the tree at level i until finding a valid leaf node at level 1. The first point found with the depth-first search SD is the Babai point (BP), which corresponds to QRD-based solution [16, 79]. Subsequently, the search continues by back-tracking to previous levels until all nodes have been solved or pruned in order to find better candidates (Figure 2.4a).

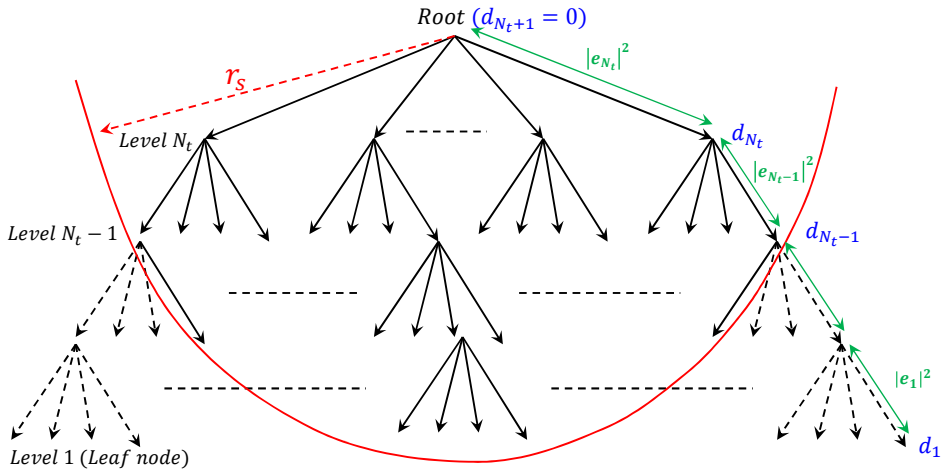


Figure 2.6: Tree-search representation of MIMO detection.

The tree-search can be also represented using a linear ZF filter with a Cholesky decomposition of Gram matrix ($\mathbf{H}^T\mathbf{H}$) instead of QR decomposition. We note that both approaches are equivalent and give the same path metrics and candidates. Through our work, QR decomposition is used to describe the tree-search problem.

Enumeration strategies

Enumeration strategy refers to the order in which the children of a node are tested. Two enumeration strategies can be used: Fincke-Pohst (FP) [17, 84] and Schnorr-Euchner (SE) [20] as represented in Figure 2.7.

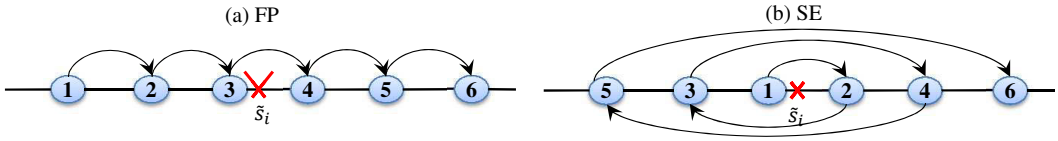


Figure 2.7: Enumeration strategies: **(a)** FP and **(b)** SE.

FP enumeration tries to find the shortest lattice vector by traveling the tree in forward and backward directions without any ordering [17]. SE enumeration was proposed as a refinement of FP enumeration by ordering the hypotheses in ascending order with respect to their Euclidean distance, where closer hypothesis will be tested first [20]. A low complexity SE decoding algorithm was therefore proposed for QAM modulation in [85]. Figure 2.7 illustrates these two enumeration strategies, where the numbers represent the order in which the hypotheses are tested.

The enumeration strategy has a major impact on the complexity of the search. Obviously, the use of SE strategy leads to reduction in the computational complexity. This is due to the fact that most probable hypothesis is first tested which reduces the number of visited nodes during the search, and avoids the computation of branch metrics for paths which will be subsequently discarded.

Layer ordering technique can be further used in preprocessing step to reduce the complexity of tree-search. It allows the selection of the most reliable symbols at a high layer using the sorted QR (SQR) decomposition [14]. The most reliable symbols are helpful for faster finding the ML solution. MMSE preprocessing might be also used for further reduction through the use of an extended channel matrix for the SQR decomposition [15].

Radius choice and tree pruning criteria

One important challenge of the sphere decoder is the choice of an initial value of the search radius r_{s}. Clearly, if r_{s} is chosen too large, the number of visited nodes may be very high and then the complexity will be increased in an exponential manner. Whereas if r_s is chosen too small, there may be no nodes inside the hypersphere.

A simple approach consists in increasing radius search (IRS) as proposed in [78, 86]. In this case, the radius is first initialized to a fixed value r_0. If no candidate is found, the search must be repeated using a larger radius ($r_1 > r_0$) which dramatically increases the detector latency. In [87] an improved increasing radius search algorithm was proposed. This algorithm exploits the most promising candidates in the incomplete tree when the search fails in order to avoid the redundant computation of branch metrics for the starting search.
Therefore, the use of a fixed sphere radius is not efficient for practical systems [80]. The efficient solution for the initial radius choice is to use an adaptive approach. It consists in initializing the radius with an infinite value, and updating it whenever a valid leaf node has been reached [16].

In the tree-search, when the partial Euclidean distance of a given node exceeds the search radius, this node is pruned. Several tree pruning techniques have been proposed to reduce the complexity of sphere decoder. In [18], it is suggested to use the Euclidean distance of the ZF solution as a pruning criterion. An increased radius algorithm (IRA) was proposed in [88]. This algorithm uses a radius with a pruning probability for each layer. In [89, 90], a statistical tree pruning approach was proposed. This method uses a probabilistic noise constraint to tighten the necessary condition on each layer.
Figure 2.4a shows an example of SD for $N_t = 2$, where solid lines and dash lines represent the forward and backward search in the tree, respectively.

It has been shown in [80, 86] that the sphere decoder achieves quasi-ML performance with polynomial average computational complexity (generally cubic) in terms of the number of transmit antennas. However, the worst case presents an exponential complexity [91]. From an implementation point of view, the SD has two main drawbacks. Firstly, its variable complexity which depends on the noise level and the channel conditions making it unsuitable for constant rate applications. Secondly, the sequential nature of the tree-search limits the performance and the level of parallelism in hardware implementation.

The first VLSI implementation of the SD algorithm was described by Burg et al. in [92], in which a parallel structure with one node per cycle is proposed. It has been demonstrated that SD is suitable for high performance hard-decision MIMO detection in practical systems.

Although sphere decoder offers significant reduction in complexity compared to ML decoder, it still requires considerable computational complexity. In order to reduce the computational complexity of SD and to obtain a constant throughput, other implementation strategies and sub-optimal algorithms have been developed such as K-Best decoder [18] and Fixed sphere decoder [93].

2.4.2 Metric-first search - Sequential decoding

The metric-first search [94] is based on stack algorithm, in which the symbol with minimum metric is extended to all its children nodes as shown in Figure 2.4b. This algorithm uses a stack to keep track of several paths simultaneously during the tree-search. The paths with best metrics are hence ordered on the top of the stack [95, 96]. The tree-search is constructed by repeatedly extracting the path from the stack which currently has the best metric and has not yet reached full length and extending it to all its children nodes. The resulting paths are stored back on the stack, the stack is then sorted. This process is repeated until the symbol vector with minimum distance is found. This algorithm is highly impacted by the size of the stack and the sorted algorithm. The search is ended once a predefined number of full length paths has reached the top of the stack.

2.4.3 K-Best decoder

K-Best algorithm [18] is based on breadth-first search in which the tree is traversed only in the forward direction. This approach commonly denoted as M-algorithm constructs the tree layer by layer retaining only a fixed number K of paths with best metrics at each detection layer. Figure 2.4c shows an example of the tree-search with $N_t = 2$. The algorithm starts by extending the root node to all possible candidates. It then sorts the new paths according to their metrics and retains the best K paths with smallest metrics for the next detection layer. It can be summarized as follows:

1. for layer $i = N_t$ to 1 Do
2. Extend each survivor path to all $\sqrt{(2^Q)}$ possible paths
3. Update the PED metric for each path
4. Sort the paths according to their PED metrics
5. Select K best paths and updates the path history accordingly
6. if layer = 1, stop the algorithm else go to step 2.

K-Best algorithm is able to achieve near optimal performance with a fixed complexity and suitable level for parallel implementation. This fixed complexity depends on the number K of retained candidates, on the size of modulation and on the number of transmit antenna, where the number of visited nodes in the tree is equal to $2^Q + (N_t - 1)K2^Q$.
However, K-Best algorithm does not take into consideration the noise variance and channel conditions. In addition, the expansion and the sorting operations are two main drawbacks of K-Best algorithm.

K-Best algorithm expands each K retained paths to its 2^Q possible children at each level. Thus, a high complexity is required to enumerate the children nodes especially in the case of high-order modulation and high number of survival paths. For this reason, several enumeration schemes have been proposed in complex domain to avoid the full expansion such as phase shift keying (PSK) enumeration, relaxed K-Best enumeration [97] and on demand expansion [98, 99]. Meanwhile, in real signal model, the enumeration can be done through a slicing operation to the nearest constellation point or simply through the use of a LUT [100]. Recognizing the low efficiency of M-algorithm with high-order modulation, multi-level enumeration methods have been proposed in [101]. This approach partitions the constellation into different sub-segments such that each layer is effectively divided into sub-layers.

Furthermore, the algorithm requires to compute and sort $2^Q K$ path metrics at each level of which $K(2^Q - 1)$ belonging to paths are pruned from the tree. Such a sorting operation is very time consuming. Several proposals have been drawn in the literature to approximate the sorting operations such as relaxed sorting [97], local sorting and merging, and distributed sorting [102], or even to avoid sorting using on demand expansion scheme [98, 103]. This scheme is independent of the constellation size and scales linearly with the value of K.

Moreover, the algorithm is prone to error propagation especially for low values of K. One way of tracking this problem is to use an adaptive value of K as a function of the tree depth [104, 105]. A large value of K is used for the first layers which is then reduced when detecting the last layers.
The first implementation of K-Best decoder is presented in[18] for 4×4 16-QAM MIMO system. Different VLSI implementations have been subsequently proposed in the literature to improve the algorithm performance [29, 102, 106, 107, 108].

2.4.4 Fixed complexity sphere decoder

The fixed sphere decoder (FSD) is another sub-optimal MIMO detection scheme proposed by Barbero et al. to further reduce the complexity of K-Best decoder [19, 109, 110, 93]. The FSD performs two stages of tree-search as illustrated in Figure 2.4d:

- Full expansion: A full expansion is performed at first p top levels, where all possible candidates are retained to the following detection levels.
- Single expansion: A single search is performed in the remaining $(N_t - p)$ levels, where only one candidate per node having lowest metric is considered for next layers.

The parameter p must be chosen such as $(N_r - N_t)(p+1) + (p+1)^2 > N_r$ in order to achieve an asymptotical ML performance and a full diversity in MIMO systems [111]. We note that in FSD, the columns of $\mathbf{H}$ must be ordered such as in the first p levels, the signal has the largest post-processing noise amplification. Meanwhile, in the remaining $(N_t - p)$ levels, signals are sorted based on their reliability where signals with least noise amplification are detected first. In conventional FSD, V-BLAST signal sorting is used to determine the order in which signals are detected. In [112], a low-complexity ordering scheme is proposed by embedding the sorting stage in the QR decomposition of the channel matrix.

The conventional FSD has a fixed complexity however it does not take into consideration the noise and channel conditions. In [113], a simplified version of the FSD has been proposed by introducing the path selection of the remaining levels. FSD algorithm can be highly parallelized and fully pipelined. Several implementations of FSD have been reported in [109, 93, 114].

2.5 Lattice reduction aided detection

The basic idea behind lattice reduction (LR) is to transform the detection problem into a better conditioned domain using a reduced lattice basis instead of the original lattice basis generating the same lattice points; thereby realizing decision regions much closer to those of the ML detector [115, 116, 117].

Several lattice reduction algorithms have been proposed in the literature with different levels of performance and complexity [16]. Minkowski and Korkine-zolotareff algorithms present optimal performance with exponential complexity [118]. As an alternate, LLL algorithm [119] and Seysen's algorithm[120] have been proposed with a polynomial complexity and have been widely considered for real and complex LR-aided detectors [117, 121]. A detailed comparison in terms of performance and complexity of these LR techniques in the context of MIMO systems have been conducted in [122, 123].

With LR technique, the original basic $\mathbf{H}$ is transformed into a new reduced basic $\tilde{\mathbf{H}}$ (more orthogonal) generating an identical lattice $\mathcal{L}(\mathbf{H}) = \mathcal{L}(\tilde{\mathbf{H}})$:

$$\tilde{\mathbf{H}} = \mathbf{HT} \tag{2.20}$$

where $\mathbf{T}$ is an unimodular matrix of dimensions $N_t \times N_t$ with integer entries and determinant ($|\det(\mathbf{T})| = 1$) [16].
Using the reduced channel matrix $\tilde{\mathbf{H}}$, the system model can be then written as [117]:

$$\mathbf{y} = \mathbf{Hs} + \mathbf{n} = \tilde{\mathbf{H}}\mathbf{z} + \mathbf{n}, \tag{2.21}$$

where $\mathbf{z} = \mathbf{T}^{-1}\mathbf{s}$. The detection is therefore performed using the equivalent system model in equation (2.21) in order to obtain an estimate of $\mathbf{z}$, $\hat{\mathbf{z}}$. Then an estimate of the transmitted vector $\mathbf{s}$, $\hat{\mathbf{s}}$, is calculated using the relationship $\tilde{\mathbf{s}} = \mathbf{T}\,\hat{\mathbf{z}}$ as illustrated in Figure 2.8.

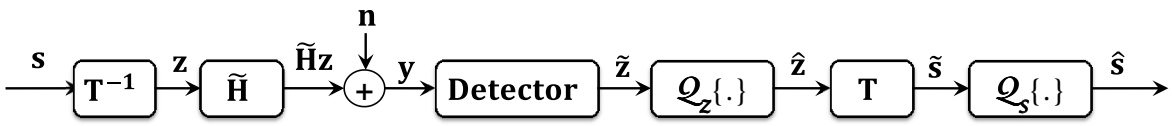


Figure 2.8: Block diagram of LR detector.

LR relaxes ML detection problem to a closed vector point (CVP) on the infinite lattice. Let $\mathcal{A}$ denotes the finite subset of real-valued transmitted signals. For M_c-QAM, this set is given by: $\mathcal{A} = \left\{\pm 1a, \pm 3a, ..., \pm(\sqrt{M}-1)a\right\}$, where $a = \sqrt{\frac{3}{2(M-1)}}$ is a power normalization coefficient ($a = 1/\sqrt{2}$,$1/\sqrt{10}$ and $1/\sqrt{42}$ for 4-QAM,16-QAM and 64-QAM respectively). The finite integer subset $\mathcal{A}^m$ can be interpreted as a shifted and scaled version of an infinite integer subset $\mathcal{Z}^m \subset \mathbb{Z}^m$ [117]:

$$\mathcal{A}^m = 2a(\mathcal{Z}^m + \frac{1}{2}\mathbf{1}_m) \tag{2.22}$$

The transformed transmitted vector $\mathbf{z}$ can be represented by:

$$\mathbf{z} = \mathbf{T}^{-1}\mathbf{s} = 2a\mathbf{T}^{-1}(\overline{\mathbf{s}} + \frac{1}{2}\mathbf{1}_m) = 2a(\overline{\mathbf{z}} + \frac{1}{2}\mathbf{T}^{-1}\mathbf{1}_m), \tag{2.23}$$

where $\overline{\mathbf{z}} = \mathbf{T}^{-1}\overline{\mathbf{s}} \in T^{-1}\mathcal{Z}^m \subset \mathbb{Z}^m$. In order to find an estimate of $\mathbf{z}$, a scaled and shifted version of $\tilde{\mathbf{z}}$ is quantized with respect to $\mathbb{Z}^m$ and re-scaled and re-shifted again:

$$\hat{\mathbf{z}} = 2a\left(\mathcal{Q}_{\mathbb{Z}}\left\{\frac{1}{2a}\tilde{\mathbf{z}} - \frac{1}{2}\mathbf{T}^{-1}\mathbf{1}_m\right\} + \frac{1}{2}\mathbf{T}^{-1}\mathbf{1}_m\right). \tag{2.24}$$

The transmitted vector $\hat{\mathbf{s}}$ can be finally obtained by the quantization of $\tilde{\mathbf{s}} = \mathbf{T}\hat{\mathbf{z}}$:

$$\hat{\mathbf{s}} = \mathcal{Q}\left\{\tilde{\mathbf{s}}\right\} = 2a\mathbf{T}\mathcal{Q}_{\mathbb{Z}}\left\{\frac{1}{2a}\tilde{\mathbf{z}} - \frac{1}{2}\mathbf{T}^{-1}\mathbf{1}_m\right\} + a\mathbf{1}_m. \tag{2.25}$$

Lattice reduction is in general combined with other low complexity detection algorithms to improve their performance at an expense of additional level of computational complexity. Linear detection and interference cancellation based detection can be easily used with LR, since a trivial quantization is required at each layer. Meanwhile, in the case of tree-search methods, the application of LR is not straightforward due to the modification of the constellation set in the reduced domain. The reduced domain neighborhood has been widely addressed in the literature, more details can be found in [124] and in the references therein.

2.6 Performance results and discussion

2.6.1 Simulation parameters

In this section, we compare the uncoded BER performance of the most prominent hard-decision detection algorithms. The simulations are based on a 4×4 SM MIMO system, QAM constellation with Gray mapping, and Rayleigh fading channel model. The Rayleigh fading coefficients are randomly generated with zero mean and unit variance. Table 2.1 summarizes the principle parameters for the simulations. The performance is measured in terms of bit error rate (BER) with respect to SNR per information bit E_b/N_0 defined by:

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} + 10\log_{10}\frac{1}{QN_t}. \quad [dB] \tag{2.26}$$

2.6.2 BER performance

Figure 2.9 compares the performance of linear detectors in a 4×4 MIMO system using several modulation orders. We show that the ML detector is obviously the optimum one and achieves a full diversity order equal to $N_r = 4$. The sphere decoder is used as an approximate of the ML solution in the case of 16-QAM and 64-QAM. MMSE detector shows better BER performance than ZF but both show a same diversity order equal to one. Moreover, the gain of MMSE over ZF is reduced with a high-order modulation (64-QAM). In general, linear detectors present significant performance loss compared to the ML detector.

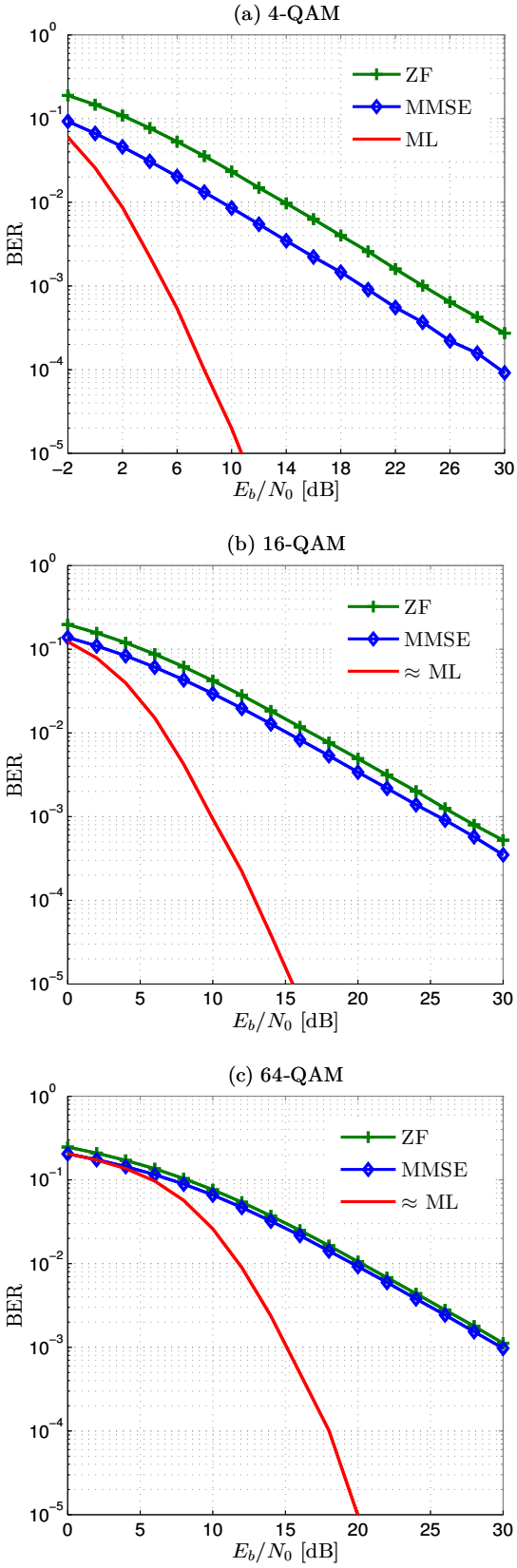


Figure 2.9: BER Performance of a $\mathbf{4 \times 4}$ uncoded MIMO system of linear detectors using 3 constellations: **(a)** 4-QAM, **(b)** 16-QAM, **(c)** 64-QAM.

Table 2.1: Simulation parameters.

MIMO system	4×4 Spatial multiplexing
Modulation 2^Q-QAM	(4,16,64)-QAM Gray mapping
Channel type	Rayleigh fading
Detector	ZF MMSE
	SIC-ZF SIC-MMSE OSIC-ZF OSIC-MMSE SQRD
	SD K-Best FSD
Channel decoder	$R_c = 1$ uncoded

Figure 2.10 shows BER performance of SIC detectors. SIC detectors are either based on V-BLAST algorithm with or without ordering (SIC or OSIC), or based on SQRD algorithm (SQRD). We view that SIC detectors achieve better performance compared to linear detectors in Figure 2.9, but still show significant performance degradation in the high SNR compared to ML detector. We show also that the symbol ordering leads to an improvement of BER performance for both SIC-ZF and SIC-MMSE algorithms. This improvement is better in the case of SIC-MMSE which indicates less error propagation compared to SIC-ZF. Furthermore, it is interesting to note that with the increase of modulation order, the improvement is reduced. Moreover, BER performance of SQRD and SQRD-MMSE based detectors is depicted in the case of 4-QAM and 64-QAM. Obviously SQRD-MMSE has better performance than SQRD. As SQRD-MMSE does not assure the optimal order, a performance gap between SQRD-MMSE and OSIC is observed.
Despite of layer ordering, none of these algorithms achieves full diversity order. Their diversity order lies between $N_r - N_t + 1 = 1$ and $N_r = 4$ and converges approximately to one for high SNR. SQRD-based detection has much lower computational complexity than the V-BLAST algorithm with a tolerable degradation in BER performance especially in case of high-order modulation.

In Figure 2.11, BER performance of tree-search detection algorithms is depicted. We show that the SD achieves ML performance with a full diversity order equal to $N_r = 4$. K-Best algorithm achieves near optimum performance depending on the K value and on the constellation size. In the case of 4-QAM, $K = 4$ is sufficient. However, with high-order constellation, a large number of retained candidates K is required in order to approach the optimum performance, i.e $K = 16$ for 16-QAM and $K = 32$ for 64-QAM. Furthermore, the performance of FSD is also depicted. We show that ordering has a crucial impact on its performance. Without signal ordering, the performance

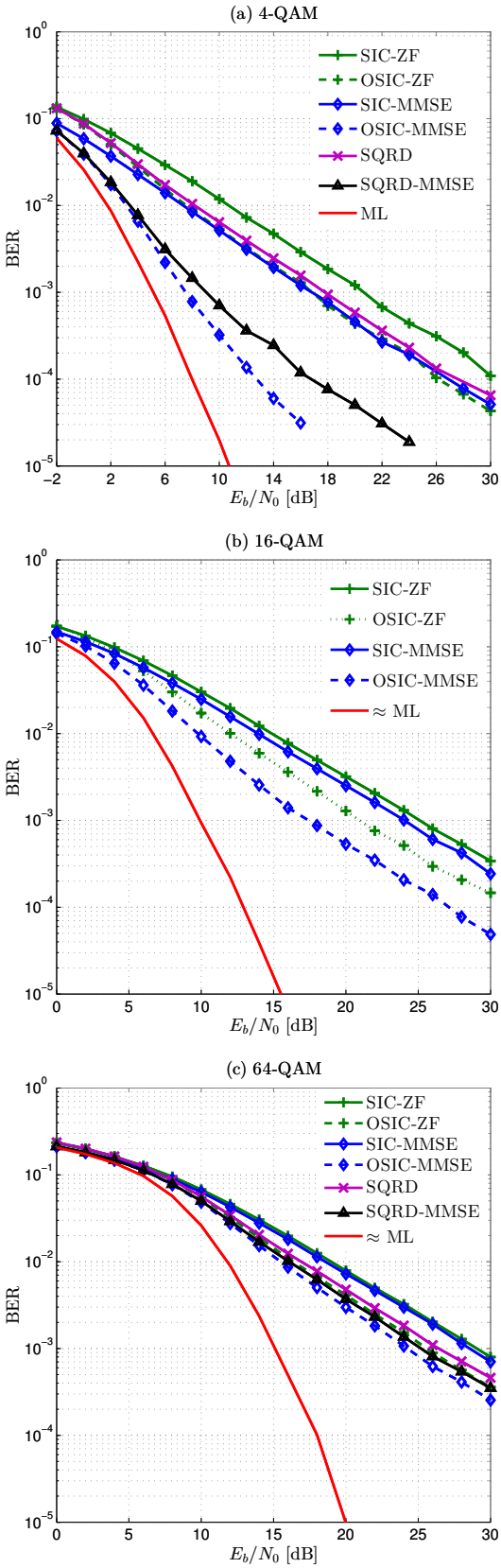


Figure 2.10: BER Performance of a $\mathbf{4 \times 4}$ uncoded MIMO system of SIC detectors using 3 constellations: **(a)** 4-QAM, **(b)** 16-QAM, **(c)** 64-QAM.

of FSD is significantly degraded and the diversity order is lower than SD. With an appropriate signal ordering, FSD is able to achieve the same diversity of SD with a small degradation in performance.

Table 2.2 summarizes the performance feature of various detectors with uncoded transmission. ZF, MMSE and SIC provide only $N_r - N_t + 1$ order diversity with various SNR losses [125]. OSIC due to the ordering process may realize more than $N_r - N_t + 1$ order diversity. However SD, K-Best and FSD are able to achieve the full diversity order N_r but with more computational complexity.

Table 2.2: Performance of hard-decision MIMO detectors.

Detector	Diversity order	Performance loss
ML	N_r	Optimal
ZF	$N_r - N_t + 1$	Very high
MMSE	$\approx N_r - N_t + 1$	high
SIC	$\approx N_r - N_t + 1$	Low
OSIC	$> N_r - N_t + 1 < N_r$	Low
SD	$\approx N_r$	$\approx$ Optimal
K-Best ($K \nearrow$)	$\approx N_r$	Near optimal
FSD	$\approx N_r$	Near optimal

2.7 Conclusion

In this chapter, we have investigated different MIMO detection algorithms including linear detection, interference cancellation and tree-search based detection. Their associated advantages and drawbacks have been presented and discussed. We have finally compared their performance with different modulations.
Until now, only hard-decision MIMO detection is considered in which the detector delivers a hard estimates of the transmitted symbols (Uncoded system). However, in the case of channel coding, the performance of the system can be further improved by using soft-decision values. These soft information can be iteratively exchanged in order to achieve near capacity. Studying the soft-input soft-output detection and analyzing the convergence behavior of the iterative receiver will be the topic of the next chapter.

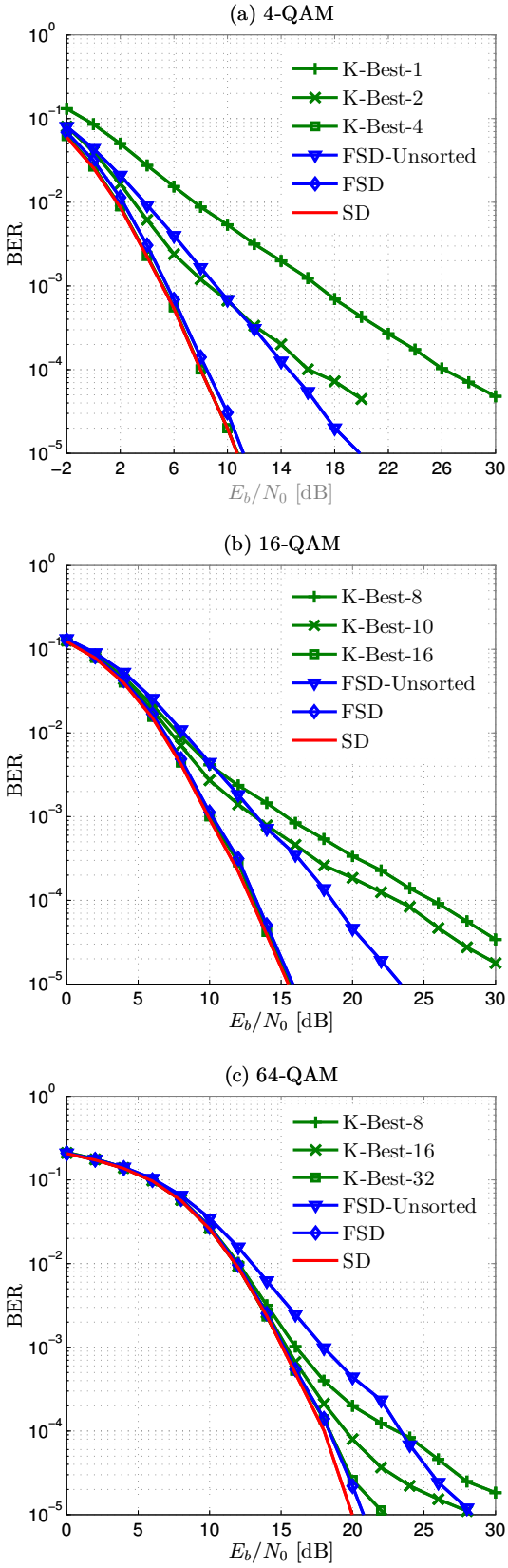


Figure 2.11: BER Performance of a $\mathbf{4 \times 4}$ uncoded MIMO system of tree-search based detectors using 3 constellations: **(a)** 4-QAM, **(b)** 16-QAM, **(c)** 64-QAM.

Chapter 3

Soft-input soft-output MIMO detection

Contents

In the previous chapter, we discussed hard-decision MIMO detection techniques. However, in the case of coded MIMO system, the performance can be improved by iteratively exchanging soft information between the MIMO detector and the channel decoder. In this chapter, we firstly describe the concept of iterative MIMO detection-decoding. We then present an overview of the main existing soft-input soft-output MIMO detection algorithms. Consequently, a low-complexity K-Best (LC-K-Best) decoder is proposed [30]. The proposed algorithm can achieve near-optimum performance while reducing the computational complexity. Moreover, we analyze the convergence behavior of the iterative receiver. The EXIT charts are adopted for a thorough analysis of the convergence behaviors of the system. Based on this analysis, a new scheduling order of the number of iterations for the iterative process is presented for an efficient trade-off between performance and complexity. At the end, simulations are carried out to validate the performance of our proposed approaches.

3.1 Iterative detection-decoding principle

In a coded MIMO-OFDM systems, the optimal way to decode the received signal would be the joint detection and decoding that reveals to be very complex and infeasible for practical implementation even for short code-blocks [42]. An alternative solution is to perform the detection and the decoding steps iteratively with soft information exchanging. Such a method, commonly referred to as iterative or turbo processing, was initially proposed for turbo decoding [2] where two decoders exchange soft information to improve the system performance. The turbo principle was rapidly extended to the turbo equalization [21], where maximum likelihood (ML) equalization and channel decoding were performed iteratively to overcome inter-symbol interference (ISI). The ML detector has been then replaced with a soft interference canceler based on linear filters with low computational complexity in [126, 127, 22, 128]. Turbo equalization principle has been subsequently applied to several transmission systems, such as systems with multi-user interference [129] and multi-antenna interference [130]. It was demonstrated in [27, 131] that iterative MIMO decoding can achieve full potential performance. However, iterative soft-input soft-output (SISO) detection algorithms exhibit higher computational complexity compared to non-iterative algorithms.

Recently, many efforts have been made in the design of soft-input soft-output MIMO detectors in order to achieve high throughput and low computational complexity. An improved VBLAST (I-VBLAST) for SISO detection was proposed in [132, 23]. In addition, SISO detector based on MMSE interference cancellation (MMSE-IC) was proposed in [24, 25]. The list sphere decoder (LSD) was proposed in [27] as a variant of the sphere decoder to provide soft-outputs. Consequently, a list sequential decoder based on metric-first search strategy was proposed for the iterative process in [95]. The single tree-search (STS) algorithm [133, 28] was then proposed to find the MAP hypothesis and the corresponding counter hypotheses during one tree-search process. In [134], the tuple search detector (TSD) was introduced to improve the trade-off between STS-SD and LSD. Furthermore, soft versions of K-Best decoder and FSD decoder for iterative receiver were proposed in [101, 29, 69] and [135, 136], respectively. It is therefore interesting to investigate MIMO detectors in order to develop a detector for a good tradeoff between performance and complexity in modern wireless communication systems.

The implementation of MIMO detectors have also been widely discussed in the literature. In [25], an implementation of SISO detector based on MMSE-IC algorithm was presented. However, this algorithm is not able to fully exploit the spatial diversity of MIMO system. Several implementations of SISO STS-SD were then reported in [137, 138] to exploit the spatial diversity. A VLSI architecture of TSD was proposed in [134]. Their main issue is their prohibitive worst-case complexity. In [139], a trellis-search-based SISO decoder and its VLSI architecture have been proposed. Such a trellis-based decoder provides a high throughput at the cost of a large hardware area. Several implementations of SISO FSD were proposed in [140, 136, 141, 142]. Several implementations of K-Best decoder were also reported [143, 144, 99]. Despite these efforts, it is still very challenging to develop a high speed iterative receiver with efficient MIMO detector to meet the high throughput requirements at an affordable complexity and implementation cost.

The iterative detection-decoding process based on turbo principle is illustrated in Figure 1.15. The MIMO detector and the channel decoder can be viewed as serially concatenated blocks. The MIMO detector can employ MAP algorithm or other sub-optimal algorithms like LSD, STS-SD, K-Best decoder, or MMSE-IC. When MIMO equalization is performed, the iterative process is referred to as turbo equalization [126, 127, 128]. The MIMO detector takes the received symbol vector $\mathbf{y}$ and the *a priori* information L_{A1} of the coded bits from the channel decoder and computes the extrinsic information L_{E1}. This extrinsic information is de-interleaved and serves as *a priori* information L_{A2} for the channel decoder. The channel decoder computes extrinsic information L_{E2} which is consequently re-interleaved and fed back to the MIMO detector as *a priori* information L_{A1}. The channel decoder can be a turbo decoder or an LDPC decoder which performs itself an iterative decoding based on log-MAP in the case of turbo code or SPA in the case of LDPC. In our iterative process, we denote by I_{out} the number of outer iterations between the MIMO detector and the channel decoder and by I_{in} the number of inner iterations within the channel decoder.

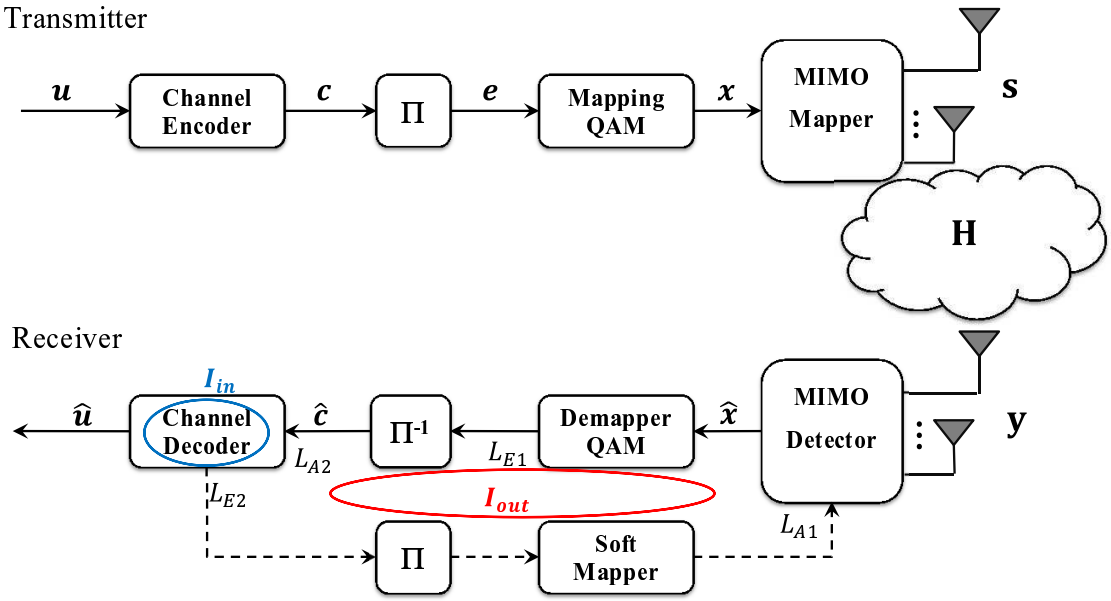


Figure 1.14: MIMO system block diagram using bit-interleaved coded modulation with iterative detection and decoding (Chapter 1).

Typically, the error rate performance of an iterative process can be divided into three regions as shown in Figure 3.1. The region of low E_b/N_0 where the performance is poor and the iterative process is not able to bring any improvement for communication system. The waterfall region or commonly referred to as cliff region in which the error rate performance curve improves steeply with iterations, providing low error rate at moderate E_b/N_0. The floor region for moderate to high E_b/N_0 in which the curve starts to flatten at a low error rate after a few number of iterations.

3.2 Maximum *a posteriori* probability (MAP) detection

MAP algorithm achieves the optimum performance through the use of an exhaustive search over all $2^{Q \cdot N_t}$ possible symbol combinations to compute the exact *a posteriori* probability of each bit.

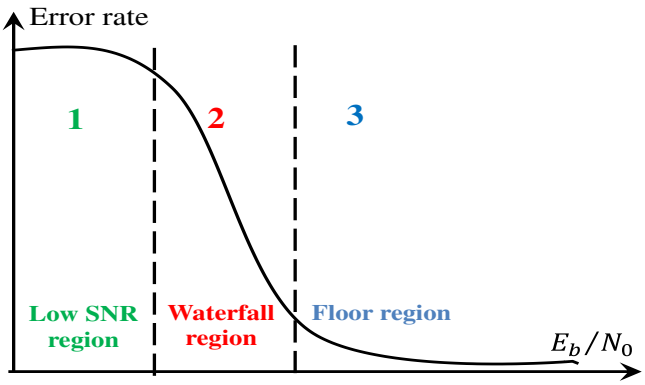


Figure 3.1: The three regions of error rate performance of an iterative process.

Such a probability is usually expressed in terms of log-likelihood ratio (LLR). The sign of LLR value determines the binary decision about the corresponding bit, while its magnitude indicates the reliability of the decision. More concretely, LLR of the b^{th} bit of the i^{th} symbol, $x_{i,b}$, can be computed as:

$$L\left(x_{i,b}\right) = \log \frac{P(x_{i,b}=+1|\mathbf{y})}{P(x_{i,b}=-1|\mathbf{y})} = \log \frac{\sum_{s\in\chi_{i,b}^{+1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}{\sum_{\mathbf{s}\in\chi_{i,b}^{-1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}, \tag{3.1}$$

where $\chi_{i,b}^{+1}$ and $\chi_{i,b}^{-1}$ denote the sets of symbol vectors corresponding to the i^{th} symbol and having the b^{th} bit of the symbol equal to $+1$ and -1 (representing a logical 1 and a logical 0), respectively. $p(\mathbf{y}|\mathbf{s})$ is the conditioned probability density function given by:

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{(\pi N_0)^{N_r}} \exp\left(-\frac{1}{N_0}\left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2\right), \tag{3.2}$$

and $P(\mathbf{s})$ represents the *a priori* information provided by the channel decoder in the form of *a priori* LLR:

$$\begin{aligned} L_A(x_{i,b}) &= \log \frac{P\left(x_{i,b}=+1\right)}{P\left(x_{i,b}=-1\right)}, \quad \forall i,b \\ P(\mathbf{s}) &= \prod_{i=1}^{N_t} P\left(s_i\right) = \prod_{i=1}^{N_t}\prod_{b=1}^{Q} P\left(x_{i,b}\right). \end{aligned} \tag{3.3}$$

To reduce the computational complexity, LLR values can be calculated using the max-log-MAP approximation [27]:

$$L\left(x_{i,b}\right) \approx \frac{1}{N_0}\min_{\chi_{i,b}^{-1}}\left\{d_1\right\} - \frac{1}{N_0}\min_{\chi_{i,b}^{+1}}\left\{d_1\right\}, \tag{3.4}$$

$$\begin{aligned} d_1 &= \left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2 - N_0 \log P(\mathbf{s}), \\ &= \left\|\mathbf{y}-\mathbf{H}\mathbf{s}\right\|^2 - N_0 \sum_{i=1}^{N_t}\sum_{b=1}^{Q} \log P(x_{i,b}), \end{aligned} \tag{3.5}$$

where d_1 represents the Euclidean distance between the received vector $\mathbf{y}$ and lattice points $\mathbf{H}\mathbf{s}$.

Note that $P(\mathbf{s})$ is obtained from the *a priori* LLR:

$$-\log P\left(s_i\right) = -\sum_{b=1}^{Q} \log P(x_{i,b}) = \tilde{K}_i - \sum_{b=1}^{Q} \frac{1}{2} x_{i,b} L_A\left(x_{i,b}\right), \tag{3.6}$$

where $\tilde{K}_i = \sum_{b=1}^{Q} \left(\frac{1}{2} \left| L_A\left(x_{i,b}\right) \right| + \log\left(1 + e^{-\left| L_A\left(x_{i,b}\right) \right|}\right)\right)$ is a constant term independent of $x_{i,b}$ that cancels out in the computation of the *a posteriori* LLR. This constant $\tilde{K}_i$ can be then set to zero. However setting it to zero may leads to negative branch metrics. To avoid this, a constant term $K_i = \sum_{b=1}^{Q} \frac{1}{2} \left| L_A\left(x_{i,b}\right) \right|$ is used instead of neglected $\tilde{K}_i$, and $-\log P\left(s_i\right)$ can be approximated by:

$$-\log P\left(s_i\right) \approx K_i - \sum_{b=1}^{Q} \frac{1}{2} x_{i,b} L_A\left(x_{i,b}\right) \approx \sum_{b=1}^{Q} \frac{1}{2} \left(\left| L_A\left(x_{i,b}\right) \right| - x_{i,b} L_A\left(x_{i,b}\right) \right). \tag{3.7}$$

Based on the *a posteriori* LLRs $L(x_{i,b})$ and the *a priori* LLRs $L_A(x_{i,b})$, the detector computes the extrinsic LLRs as:

$$L_E(x_{i,b}) = L(x_{i,b}) - L_A(x_{i,b}). \tag{3.8}$$

The exact computation of LLR using MAP detection can only be used with low-order modulations and small number of antennas [71]. However, with high-order modulations and large number of antennas, MAP algorithm becomes unfeasible similarly to ML algorithm previously discussed. This is due to its exponential complexity since $2^{Q \cdot N_t}$ hypotheses have to be considered within each minimum term and for each bit.

3.3 Linear soft-output detection

Due to their low-complexity, linear detectors including ZF and MMSE are extended to provide soft-outputs [126, 26, 145]. MIMO detection problem is thus decoupled into a set of parallel independent single-input single-output detection problems. The LLRs can be separately computed for each layer i as [26]:

$$L\left(x_{i,b}\right) \approx \log \frac{P(x_{i,b} = +1 | y_i)}{P(x_{i,b} = -1 | y_i)}, \tag{3.9}$$

$$L\left(x_{i,b}\right) \approx \min_{s_i \in \chi_{i,b}^{-1}} \left\{ \frac{\left| \tilde{s}_i - s_i \right|^2}{\sigma_{\eta_i}^2} - \log P(s_i) \right\} - \min_{s_i \in \chi_{i,b}^{+1}} \left\{ \frac{\left| \tilde{s}_i - s_i \right|^2}{\sigma_{\eta_i}^2} - \log P(s_i) \right\}, \tag{3.10}$$

where $\tilde{s}_i = \left(\mathbf{G}_{\mathrm{LD}} \mathbf{y}\right)_i$ is the equalized symbol which is generally associated with a bias factor β_i in addition to some residual noise plus interferences η_i especially in the case of MMSE.

$$\mathbf{G}_{\mathrm{LD}} = \begin{cases} \mathbf{G}_{\mathrm{ZF}} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H & \text{in the case of ZF ,} \\ \mathbf{G}_{\mathrm{MMSE}} = \mathbf{B}(\mathbf{H}^H \mathbf{H} + \sigma_n^2 \mathbf{I}_{N_t})^{-1} \mathbf{H}^H & \text{in the case of MMSE,} \end{cases}$$

where $\mathbf{B}$ is a diagonal matrix which removes the bias from the MMSE estimates [26]. In the case of ZF, the noise is colored, hence $\sigma_{\eta_i}^2$ represents the diagonal element of the

covariance matrix: $\sigma^2_{\eta_i} = \sigma^2_n \sum_{j=1}^{N_t} \left|\mathbf{G}_{\mathrm{ZF}i,j}\right|^2$. While, in the case of MMSE, $\sigma^2_{\eta_i}$ represents the colored noise in addition to the residual interference: $\sigma^2_{\eta_i} = (\mathbf{G}_{\mathrm{MMSE}i,i} - 1)/\mathbf{G}_{\mathrm{MMSE}i,i}$.

It is important to note that equation (3.10) only requires to evaluate 2^Q terms per each bit, which significantly reduces the computational complexity compared to that of MAP computation in equation (3.4). However, linear soft-output detection entails a performance loss since the resulting LLRs are no longer optimal.
The complexity of LLR computation can be further reduced when considering the Gray labelling. This allows to separate the in-phase and quadrature components which results in extremely efficient expressions for different modulation orders [26]. This aspect will be detailed in the next section.

Figure 3.2 shows BER performance of soft-output linear detectors for 4×4 spatially multiplexed MIMO system, with 4-QAM and 64-QAM in Rayleigh fading channel. LTE turbo code specified in 3GPP LTE standard was used ($(13, 15)_o$, $R_\mathrm{c} = 1/3$). Other code rates (e.g $R_\mathrm{c} = 1/2, 3/4$) are obtained through repetition or puncturing using the rate matching module. Eight iterations inside the turbo decoder are performed. It is obvious that MIMO technique in coded systems significantly attains better performance than that of uncoded systems (Figure 2.9), since the channel decoder is able to further mitigate the impact of fading and noise. We show that soft-output ZF presents significant performance degradation about 5 dB compared to MMSE in the case of 4-QAM. SO-MMSE presents a degradation of 0.5 dB compared to ML. However, this gap increases to more than 3 dB with high-order modulation (64-QAM).

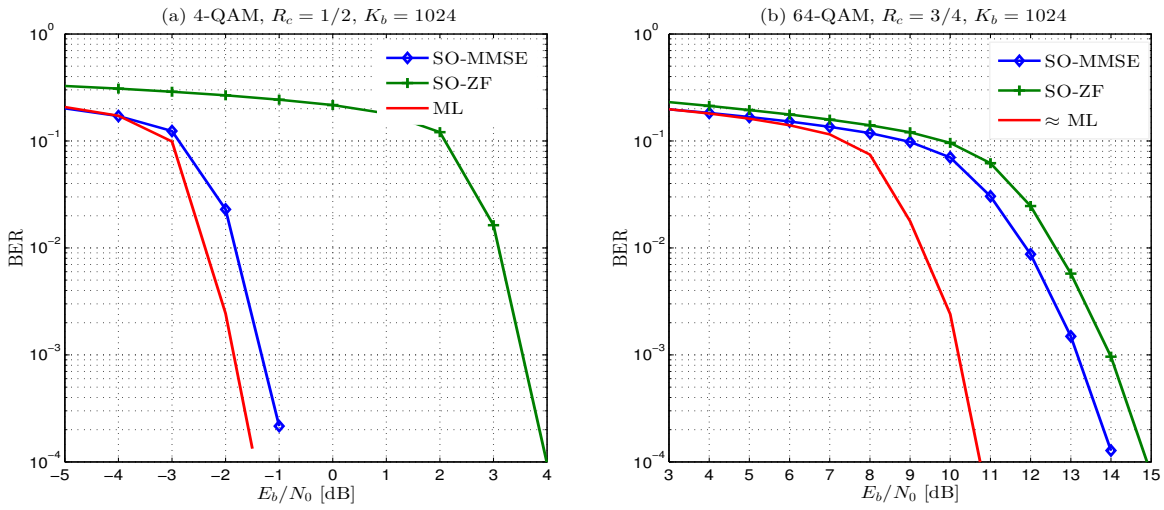


Figure 3.2: BER Performance of a $\mathbf{4 \times 4}$ coded MIMO system of soft-output LDs with: **(a)** 4-QAM, **(b)** 64-QAM.

Summarizing these observations enable us to conclude that soft-output MMSE detection is one of the most promising solutions for low-complexity detection. However, this algorithm still presents a performance loss compared to MAP detection especially with high-order modulation. Therefore, more advanced techniques are required for iterative MIMO decoding in order to substantially improve the performance of coded MIMO systems and to approach the channel capacity as will be discussed in the following sections.

3.4 Interference cancellation (IC)-based detection

Interference cancellation-based detection is commonly used in combination with MMSE linear filtering in order to reduce the interference compared to LDs as presented in section (2.3). In the case of iterative receiver, the MIMO equalizer and the channel decoder exchange soft information according to the turbo equalization principle [126, 127, 130, 128]. The MIMO equalizer produces an equalized symbol vector $\tilde{\mathbf{s}}$ deduced from received signal $\mathbf{y}$. The soft estimated symbol vector $\hat{\mathbf{s}}$ is used to cancel the interference terms in the received signal. The interference cancellation can be carried out either in a successive way as in VBLAST [13] and SQRD based methods [14] or in a parallel way as in MMSE-IC [24, 25].

3.4.1 MMSE-IC equalizer

MMSE-IC equalizer can be performed using two filters [24]. The first filter $\mathbf{p}_i$ is applied to the received vector $\mathbf{y}$, and the second filter $\mathbf{q}_i$ is applied to the estimated vector $\hat{\mathbf{s}}$ as shown in Figure 3.3.

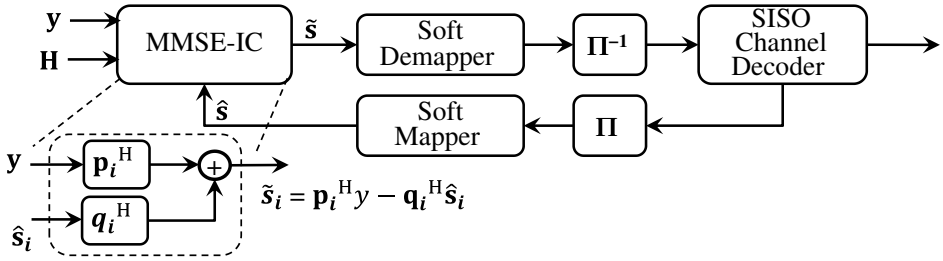


Figure 3.3: Turbo equalization with MMSE-IC.

The equalized symbol $\tilde{s}_i$ can be written as:

$$\tilde{s}_i = \mathbf{p}_i^H \mathbf{y} - \mathbf{q}_i^H \hat{\mathbf{s}}_i \quad \text{with} \quad i \in [1, N_t]\,, \tag{3.11}$$

where $\hat{\mathbf{s}}_i$ denotes the estimated vector given by the previous iteration with the i^{th} symbol omitted: $\hat{\mathbf{s}}_i = [\hat{s_1}...\hat{s}_{i-1} \quad 0 \quad \hat{s}_{i+1}...\hat{s}_{N_t}]$.
The filters $\mathbf{p}_i$ and $\mathbf{q}_i$ are optimized under MMSE criterion:

$$\left(\mathbf{p}_i^{\text{opt}}, \mathbf{q}_i^{\text{opt}}\right) = \arg\min_{\mathbf{p}_i, \mathbf{q}_i} \mathbb{E}\left\{|s_i - \tilde{s}_i|^2\right\}, \tag{3.12}$$

and can be computed by [127]:

$$\mathbf{p}_i^{\text{opt}} = \sigma_s^2 \left(\mathbf{H}\mathbf{V}_i\mathbf{H}^H + \sigma_n^2 \mathbf{I}_N\right)^{-1} \mathbf{h}_i, \tag{3.13a}$$

$$\mathbf{q}_i^{\text{opt}} = \mathbf{H}^H \mathbf{p}_i^{\text{opt}}, \tag{3.13b}$$

where σ_s^2 is the power of the received signal, $\mathbf{h}_i$ denotes the i^{th} column of the channel matrix $\mathbf{H}$, and $\mathbf{V}_i$ is a diagonal matrix that depends on the residual errors of estimated symbols:

$$\mathbf{V}_i = \sigma_s^2 \mathbf{e}_i \mathbf{e}_i^T + \sum_{j=1, j\neq i}^{N_t} \nu_j^2 \mathbf{e}_j \mathbf{e}_j^T \tag{3.14}$$

with ν_i^2 defined as:

$$\nu_i^2 = \mathbb{E}\left\{ \left| s_i - \hat{s}_i \right|^2 / L_A \right\}, \tag{3.15a}$$

$$\nu_i^2 = \sum_{s \in 2^Q} |s|^2 P\left(\hat{s}_i = s\right) - \left|\hat{s}_i\right|^2. \tag{3.15b}$$

The bias factor β_i and the variance of the noise plus interference terms $\sigma_{\eta_i}^2$ are given by:

$$\beta_i = \mathbf{p}_i^H \mathbf{h}_i \tag{3.16a}$$

$$\sigma_{\eta_i}^2 = \sigma_s^2 \beta_i \left(1 - \beta_i\right) \tag{3.16b}$$

At the first iteration, since no *a priori* information is available, the equalization process is reduced to classical MMSE solution:

$$\tilde{\mathbf{s}} = \left(\mathbf{H}^H \mathbf{H} + \frac{\sigma_n^2}{\sigma_s^2} \mathbf{I}_{N_t} \right)^{-1} \mathbf{H}^H \mathbf{y}. \tag{3.17}$$

These equalized symbols are then used by the soft demapper to compute LLR values. MMSE-IC equalizer requires N_t matrix inversions for each symbol vector and for each iteration. Hence, it entails high computational complexity which limits its implementation in practical systems. For this reason, several approximations of MMSE-IC have been proposed.

The first approximation of MMSE-IC consists in replacing the variable ν_i^2 by its mean $\nu^2 = \mathbb{E}\left\{\nu_i^2\right\}$ [128]. Hence, one matrix inversion is computed for all symbols. This approximation is denoted as MMSE-IC1. By further assuming that $\hat{s}_i$ is a random variable that contains sufficient information for the detector, ν^2 can be estimated with the power of the transmitted signal and estimated as [127, 128]:

$$\nu^2 = \mathbb{E}\left\{\nu_i^2\right\} = \sigma_s^2 - \sigma_{\hat{s}}^2. \tag{3.18}$$

With these approximations, the filtering vector $\mathbf{p}_i$ becomes:

$$\mathbf{p}_i = \sigma_s^2 \left(\mathbf{H} \left(\nu^2 \mathbf{I}_{N_t} + \sigma_{\hat{s}}^2 \mathbf{e}_i \mathbf{e}_i^T \right) \mathbf{H}^H + \sigma_n^2 \mathbf{I}_{N_r} \right)^{-1} \mathbf{h}_i \tag{3.19}$$

Using the Sherman-Morrison-Woodbury theorem, $\mathbf{p}_i$ can be written as:

$$\mathbf{p}_i = \lambda_i \bar{\mathbf{p}}_i, \tag{3.20}$$

where

$$\begin{aligned} \bar{\mathbf{p}}_i &= \left(\mathbf{H}\mathbf{H}^H \nu^2 + \sigma_n^2 \mathbf{I}_{N_r} \right)^{-1} \mathbf{h}_i, \\ \lambda_i &= \frac{\sigma_s^2}{1 + \sigma_{\hat{s}}^2 \mathbf{h}_i^H \bar{\mathbf{p}}_i} \end{aligned} \tag{3.21}$$

MMSE-IC1 algorithm reduces significantly the complexity of computing the filter coefficients. However, the coefficients of the equalizer must be recomputed at each iteration.

A second approximation denoted as MMSE-IC2 assumes a perfect estimation of transmitted symbols ($\sigma_{\hat{s}}^2 = \sigma_s^2$) to overcome the matrix inversion at each iteration [24]. Using this assumption, we get:

$$\mathbf{p}_i = \frac{\sigma_s^2}{\sigma_n^2 + \sigma_s^2 \mathbf{h}_i^H \mathbf{h}_i} \mathbf{h}_i. \tag{3.22}$$

The first FPGA implementation of iterative MIMO decoding based on MMSE-IC was proposed in [24], where a sub-optimal MMSE filter in combination with STBC scheme have been used to achieve good performance. The implementation is realized using CORDIC operators to limit the latency and the complexity of iterative process.

In [146], a novel approach of MMSE-IC has been proposed to significantly reduce the computational complexity associated with MMSE filtering matrix inversion without degrading the error rate performance. The proposed algorithm computes only a single matrix inversion for all N_t MMSE filters at once after interference cancellation. This approach is denoted as LC-MMSE-IC in the sequel. An ASIC implementation and a VLSI architecture have been then presented in [25]. This architecture employes an LU-decomposition based matrix inversion and achieves up to 600 Mbps in IEEE 802.11n.

Soft mapper

Soft mapper converts the *a priori* information coming out from the channel decoder into a soft estimated symbol. The soft estimated symbol $\hat{s}_i$ of the transmitted symbol s_i is computed according to [26]:

$$\hat{s}_i = \mathbb{E}\{s_i\} = \sum_{s \in 2^Q} sP(s_i = s), \tag{3.23}$$

where $P(s_i = s)$ corresponds to the *a priori* probability of the symbol s. Since we use a BICM scheme, the transmitted bits are assumed to be statistically independent, the *a priori* probability can be then computed as:

$$P(s_i = s) = \prod_{b=1}^{Q} P(x_{i,b}), \tag{3.24}$$

where $x_{i,b}$ is the b^{th} bit of symbol s_i having a bit value $\{-1, 1\}$, and $P(x_{i,b})$ corresponds to the probability of the bit $x_{i,b}$. $P(x_{i,b})$ can be computed using the following equations:

$$\begin{aligned} P(x_{i,b} = 1) &= \frac{e^{L_{i,b}^A}}{1 + e^{L_{i,b}^A}} = \frac{1}{2}\left(1 - \tanh\left(\frac{L_{i,b}^A}{2}\right)\right), \\ P(x_{i,b} = -1) &= \frac{1}{1 + e^{L_{i,b}^A}} = \frac{1}{2}\left(1 + \tanh\left(\frac{L_{i,b}^A}{2}\right)\right). \end{aligned} \tag{3.25}$$

The hyperbolic tangent function (tanh) can be efficiently approximated by using a look-up table (LUT) for positive part only in hardware implementation.
Using the Gray mapping, the real and imaginary parts of each symbol are independently mapped. This mapping can be exploited to reduce the computational complexity of soft estimated symbols that can be computed using the probability of the transmitted

bits with simple additions and multiplications as in Table 3.1 [26]. The variance of the soft symbol ν_i^2 can be also computed efficiently by separating the real and imaginary parts.

It was shown in [130], that using *a posteriori* LLR for soft symbol computation leads to a significant performance improvement compared to using extrinsic LLR. Therefore, the *a posteriori* LLR are considered in the remainder of this work to compute soft estimates.

Table 3.1: Soft estimated symbols for Gray mapping.

Constellation	$\mathrm{Re}\{\hat{s}_i\}$
4-QAM	$\frac{1}{\sqrt{2}}\tanh\left(\frac{L_{i,1}^A}{2}\right)$
16-QAM	$\frac{1}{\sqrt{10}}\left(\tanh\left(\frac{L_{i,1}^A}{2}\right)\right)\left(2-\tanh\left(\frac{L_{i,2}^A}{2}\right)\right)$
64-QAM	$\frac{1}{\sqrt{42}}\left(\tanh\left(\frac{L_{i,1}^A}{2}\right)\right)\left(4+\tanh\left(\frac{L_{i,2}^A}{2}\right)\tanh\left(\frac{L_{i,3}^A}{2}\right)-2\tanh\left(\frac{L_{i,2}^A}{2}\right)\right)$
Constellation	$\mathrm{Im}\{\hat{s}_i\}$
4-QAM	$\frac{1}{\sqrt{2}}\tanh\left(\frac{L_{i,2}^A}{2}\right)$
16-QAM	$\frac{1}{\sqrt{10}}\left(\tanh\left(\frac{L_{i,3}^A}{2}\right)\right)\left(2-\tanh\left(\frac{L_{i,4}^A}{2}\right)\right)$
64-QAM	$\frac{1}{\sqrt{42}}\left(\tanh\left(\frac{L_{i,4}^A}{2}\right)\right)\left(4+\tanh\left(\frac{L_{i,5}^A}{2}\right)\tanh\left(\frac{L_{i,6}^A}{2}\right)-2\tanh\left(\frac{L_{i,5}^A}{2}\right)\right)$

Soft demapper

Linear equalization transforms MIMO system into N_t single-input single-output systems assumed to be statically independent. Using this assumption and applying Bayes's theorem, the LLR can be computed per layer as:

$$L\left(x_{i,b}\right) \approx \log \frac{P\left(x_{i,b}=+1|\tilde{s}_i\right)}{P\left(x_{i,b}=-1|\tilde{s}_i\right)} \approx \log \frac{\sum_{s\in\chi_{i,b}^{+1}} P\left(\tilde{s}_i|s_i=s\right)P\left(s_i=s\right)}{\sum_{s\in\chi_{i,b}^{-1}} P\left(\tilde{s}_i|s_i=s\right)P\left(s_i=s\right)} \tag{3.26}$$

where $P\left(\tilde{s}_i|s_i=s\right)$ denotes the probability density function of $\tilde{s}_i$ assumed to be a Gaussian distribution with a mean β_i and a variance $\sigma_{\eta_i}^2$:

$$P\left(\tilde{s}_i|s_i=s\right) = \frac{1}{\pi\sigma_{\eta_i}^2}\exp\left(-\frac{\left|\tilde{s}_i-\beta_i s\right|^2}{\sigma_{\eta_i}^2}\right). \tag{3.27}$$

To reduce the computational complexity, max-log-MAP approximation can be used to compute the *a posteriori* LLRs. Furthermore, it has been shown that neglecting the *a priori* information does not have a significant performance loss. Hence, the *a posteriori* LLRs can be approximated by:

$$L\left(x_{i,b}\right) = \frac{1}{\sigma_{\eta_i}^2}\left(\min_{s\in\chi_{i,b}^{-1}}\left|\tilde{s}_i-\beta_i s_i\right|^2 - \min_{s\in\chi_{i,b}^{+1}}\left|\tilde{s}_i-\beta_i s_i\right|^2\right). \tag{3.28}$$

Using an unbiased symbol estimate $\tilde{z}_i = \beta_i^{-1}\tilde{s}_i$, the LLR can be expressed by:

$$L\left(x_{i,b}\right) = \frac{\beta_i^2}{\sigma_{\eta_i}^2}\left(\min_{s\in\chi_{i,b}^{-1}}\left|\tilde{z}_i - s_i\right|^2 - \min_{s\in\chi_{i,b}^{+1}}\left|\tilde{z}_i - s_i\right|^2\right) = \rho_i\lambda\left(\tilde{z}_i\right), \tag{3.29}$$

where

$$\rho_i = \frac{\beta_i^2}{\sigma_{\eta_i}^2} = \frac{\beta_i}{1-\beta_i}, \quad \text{and} \quad \lambda\left(\tilde{z}_i\right) = \left(\min_{s\in\chi_{i,b}^{-1}}\left|\tilde{z}_i - s_i\right|^2 - \min_{s\in\chi_{i,b}^{+1}}\left|\tilde{z}_i - s_i\right|^2\right). \tag{3.30}$$

The LLR can be efficiently computed in case of Gray mapping by simple subtraction, multiplication and comparison as summarized in Table 3.2 for unbiased MMSE equalizer [26].

Table 3.2: Low-complexity max-log LLR computation for Gray mapping.

Constellation	Bit position	$\lambda\left(\tilde{z}_i\right)$	Range
4-QAM	1	$4\,\mathrm{Re}\left(\tilde{z}_i\right)$	$\forall\,\mathrm{Re}\left(\tilde{z}_i\right)$
	2	$4\,\mathrm{Im}\left(\tilde{z}_i\right)$	$\forall\,\mathrm{Im}\left(\tilde{z}_i\right)$
16-QAM	1	$4\,\mathrm{Re}\left(\tilde{z}_i\right)$	$\lvert\mathrm{Re}\left(\tilde{z}_i\right)\rvert \leq 2$
		$8(\mathrm{Re}(\tilde{z}_i) - \mathrm{sign}(\mathrm{Re}(\tilde{z}_\mathrm{i})))$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \geq 2$
	2	$4(2 - \lvert\mathrm{Re}(\tilde{z}_i)\rvert)$	$\forall\,\mathrm{Re}(\tilde{z}_i)$
	3	$4\,\mathrm{Im}\left(\tilde{z}_i\right)$	$\lvert\mathrm{Im}\left(\tilde{z}_i\right)\rvert \leq 2$
		$8(\mathrm{Im}(\tilde{z}_i) - \mathrm{sign}(\mathrm{Im}(\tilde{z}_\mathrm{i})))$	$\lvert\mathrm{Im}(\tilde{z}_i)\rvert \geq 2$
	4	$4(2 - \lvert\mathrm{Im}(\tilde{z}_i)\rvert)$	$\forall\,\mathrm{Im}(\tilde{z}_i)$
64-QAM	1	$4\,\mathrm{Re}\left(\tilde{z}_i\right)$	$\lvert\mathrm{Re}\left(\tilde{z}_i\right)\rvert \leq 2$
		$8(\mathrm{Re}(\tilde{z}_i) - \mathrm{sign}(\mathrm{Re}(\tilde{z}_\mathrm{i})))$	$2 \leq \lvert\mathrm{Re}(\tilde{z}_i)\rvert \leq 4$
		$12(\mathrm{Re}(\tilde{z}_i) - 2\mathrm{sign}(\mathrm{Re}(\tilde{z}_\mathrm{i})))$	$4 \leq \lvert\mathrm{Re}(\tilde{z}_i)\rvert \leq 6$
		$16(\mathrm{Re}(\tilde{z}_i) - 3\mathrm{sign}(\mathrm{Re}(\tilde{z}_\mathrm{i})))$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \geq 6$
	2	$8(3 - \lvert\mathrm{Re}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \leq 2$
		$4(4 - \lvert\mathrm{Re}(\tilde{z}_i)\rvert)$	$2 \leq \lvert\mathrm{Re}(\tilde{z}_i)\rvert \leq 6$
		$8(5 - \lvert\mathrm{Re}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \geq 6$
	3	$4(\lvert\mathrm{Re}\left(\tilde{z}_i\right)\rvert - 2)$	$\lvert\mathrm{Re}\left(\tilde{z}_i\right)\rvert \leq 4$
		$4(6 - \lvert\mathrm{Re}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \geq 4$
	4	$4\,\mathrm{Im}\left(\tilde{z}_i\right)$	$\lvert\mathrm{Im}\left(\tilde{z}_i\right)\rvert \leq 2$
		$8(\mathrm{Im}(\tilde{z}_i) - \mathrm{sign}(\mathrm{Im}(\tilde{z}_\mathrm{i})))$	$2 \leq \lvert\mathrm{Im}(\tilde{z}_i)\rvert \leq 4$
		$12(\mathrm{Im}(\tilde{z}_i) - 2\mathrm{sign}(\mathrm{Im}(\tilde{z}_\mathrm{i})))$	$4 \leq \lvert\mathrm{Im}(\tilde{z}_i)\rvert \leq 6$
		$16(\mathrm{Im}(\tilde{z}_i) - 3\mathrm{sign}(\mathrm{Im}(\tilde{z}_\mathrm{i})))$	$\lvert\mathrm{Im}(\tilde{z}_i)\rvert \geq 6$
	5	$8(3 - \lvert\mathrm{Im}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Im}(\tilde{z}_i)\rvert \leq 2$
		$4(4 - \lvert\mathrm{Im}(\tilde{z}_i)\rvert)$	$2 \leq \lvert\mathrm{Im}(\tilde{z}_i)\rvert \leq 6$
		$8(5 - \lvert\mathrm{Im}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Im}(\tilde{z}_i)\rvert \geq 6$
	6	$4(\lvert\mathrm{Im}\left(\tilde{z}_i\right)\rvert - 2)$	$\lvert\mathrm{Im}\left(\tilde{z}_i\right)\rvert \leq 4$
		$4(6 - \lvert\mathrm{Im}(\tilde{z}_i)\rvert)$	$\lvert\mathrm{Re}(\tilde{z}_i)\rvert \geq 4$

3.4.2 Successive interference cancellation (SIC) equalizer

The SIC-based detector was initially used in VBLAST systems. We have seen in section (2.3.1) that in VBLAST architecture [13] a successive cancellation step and interference nulling step are used to detect the transmitted symbols. However, this method suffers from error propagation. Several methods have been proposed to reduce this problem by taking decision errors into account [147, 23]. An improved VBLAST for iterative detection and decoding is described in [147]. At the first iteration, an enhanced VBLAST which takes decision errors into account is employed. When the *a priori* LLRs are available from the channel decoder, soft symbols are computed by a soft mapper and are used in the interference cancellation for the next iterations.

To describe the enhanced VBLAST algorithm, we assume that the detection order has been made according to the optimal detection order [13]. We define $\hat{\mathbf{s}}_{i-1}$ as $[\hat{s_1} \quad \hat{s}_2 \quad ... \quad \hat{s}_{i-1}]$, and $\mathbf{H}_{i:j}$ as $[\mathbf{h}_i \quad \mathbf{h}_{i+1} \quad ... \quad \mathbf{h}_j]$, where $\mathbf{h}_i$ denotes the i^{th} column of $\mathbf{H}$. At step i, the pre-detected symbol vector $\hat{s}_{i-1}$ until step $i-1$ is canceled out from the received signal:

$$\mathbf{y}_i = \mathbf{y} - \mathbf{H}_{1:i-1}\hat{\mathbf{s}}_{i-1}. \tag{3.31}$$

In the conventional VBLAST algorithm, the hard estimated symbol vector $\mathbf{s}_{i-1}$ is used in the cancellation step, MMSE filtering is then applied in the nulling step. The enhanced VBLAST algorithm uses the soft estimated symbol vector $\hat{\mathbf{s}}_{i-1}$ and a nulling matrix $\mathbf{W}_i$ based on MMSE criterion that takes decision errors into account. $\mathbf{W}_i$ can be expressed by [147, 23]:

$$\mathbf{W}_i = \sigma_s^2 \left(\mathbf{H}\Sigma_i\mathbf{H}^H + \sigma_n^2\mathbf{I}_{N_r}\right)^{-1}\mathbf{h}_i, \tag{3.32}$$

where Σ_i is the decision error covariance matrix defined by:

$$\Sigma_i = \sum_{j=1}^{i-1}\epsilon_j^2\mathbf{e}_j\mathbf{e}_j^T + \sum_{j=i}^{N_t-i+1}\sigma_s^2\mathbf{e}_j\mathbf{e}_j^T, \quad \epsilon_j^2 = \mathbb{E}\left\{\left|s_j - \hat{s}_j\right|^2/\hat{s}_{j-1}\right\}. \tag{3.33}$$

The estimated symbol $\tilde{s}_i$ can be expressed as:

$$\tilde{s}_i = \mathbf{W}_i^H\mathbf{y}_i = \beta_i s_i + \eta_i. \tag{3.34}$$

A soft demapper is then used to compute the LLRs in equation (3.28). We refer to this algorithm as improved VBLAST (I-VBLAST) in the sequel.

3.4.3 SQRD-based detection

Similarly to V-BLAST algorithm, QRD-based algorithm has been extended to allow an iterative computation of LLR. LLR can be therefore computed as [148]:

$$L\left(x_{i,b}\right) \approx \frac{1}{\sigma_{SQRDi}^2}\left(\min_{s\in\chi_{i,b}^{-1}}\left|\tilde{y}_i - \mathbf{R}_{i,i}s_i - \sum_{j=i+1}^{N_t}\mathbf{R}_{i,j}\tilde{s}_j\right|^2 - \min_{s\in\chi_{i,b}^{+1}}\left|\tilde{y}_i - \mathbf{R}_{i,i}s_i - \sum_{j=i+1}^{N_t}\mathbf{R}_{i,j}\tilde{s}_j\right|^2\right), \tag{3.35}$$

where $\tilde{s}_j$ are the soft-estimated symbols and σ^2_{SQRDi} is the residual noise plus interference power given by:

$$\sigma^2_{SQRDi} = \sigma^2_n + \sum_{j=i+1}^{N_t} \left\{ |\mathbf{R}_{i,j}|^2 \operatorname{var}\{\tilde{\mathrm{s}}_\mathrm{j}\} \right\}. \tag{3.36}$$

It has been shown in [147], that SQRD-based detection is very sub-optimal especially for high-order modulation due to error propagation. Therefore, this solution is not discussed in the rest of the present work.

3.5 Soft-input soft-output tree-search-based detection

Viewing that sphere decoding achieves near optimum performance in hard-decision system, its principle is extended to support soft-decision. Various soft-input soft-output algorithms based on sphere decoder principle have been widely studied in the literature. The most prominent ones are briefly outlined below.

3.5.1 List sphere decoder

As previously seen, the Euclidean distance metric d_1 in equation (3.5) can be recursively evaluated through the accumulated partial Euclidean distance d_i with $d_{N_t+1} = 0$ as follows [28]:

$$d_i = d_{i+1} + \underbrace{\left| \tilde{y}_i - \sum_{j=i}^{N_t} R_{i,j} s_j \right|^2}_{m_i^C} + \underbrace{\frac{N_0}{2} \sum_{b=1}^{Q} \left(|L_A(x_{i,b})| - x_{i,b} L_A(x_{i,b}) \right)}_{m_i^A}, \quad i = N_t, ..., 1. \tag{3.37}$$

where m_i^C and m_i^A denote the channel-based partial metric and the *a priori*-based partial metric at the i^{th} level, respectively.

List sphere decoder is a modification of sphere decoder initially proposed in 2003 by Hochwald and Ten Brink [27]. LSD generates a list $\mathcal{L} \subset 2^{QN_t}$ that includes the best possible hypotheses for which the metric is small. The LLR values are then computed from this list:

$$L(x_{i,b}) = \frac{1}{N_0} \min_{\mathcal{L} \cap \chi^{-1}_{i,b}} \{d_1\} - \frac{1}{N_0} \min_{\mathcal{L} \cap \chi^{+1}_{i,b}} \{d_1\}. \tag{3.38}$$

The list is obtained using an administration list and a modified radius reduction as follows: when a leaf node inside the sphere radius is found, this point is added to the list. If the list is full, this point is compared to the point of the list having the largest distance. If this point has smaller distance, it replaces the largest distance point in the list. The search radius is then updated to the largest distance in the list. A modification of SD algorithm to handle complex constellation was also proposed [27].

It was shown that the proposed iterative detection and decoding process is able to achieve near-capacity in MIMO systems. However, the proximity to the capacity depends on the list size. The list should be large enough to include at least one candidate for both possible hypotheses. However, using an excessively large list size will lead to an increase in the computational complexity. Meanwhile, the size of the

list should not be too small either. The use of limited list size causes inaccurate approximation due to missing some counter hypotheses where no entry can be found in the list for a particular bit $x_{i,b} = \{+1, -1\}$, the LLR value is then infinity. This issue is known as missing counter-hypothesis problem.
Several solutions have been proposed to handle this issue as will be discussed later. LLR clipping is the frequently used solution which simply consists in setting the LLR to a predefined maximum value [27, 101].

Moreover, two methods were used to process the list in the iterative receiver. The first method consists in generating the list during the first iteration and using this list for subsequent iterations to update the soft information [27]. The second method updates the list at each iteration taking the *a priori* LLRs into consideration leading to further performance improvement, but yielding an additional computational complexity [149, 101].

Additionally, several methods previously introduced can be included for further reduction of the complexity of tree-search algorithms namely the Schnorr-Euchner (SE) enumeration [20], layer ordering technique and MMSE preprocessing. However, MMSE preprocessing method introduces a biasing factor (self-interference) in the metrics which should be removed in LLR calculation to avoid performance degradation as described in [150].

Different algorithms related to LSD follows in the literature. In [151], the proposed decoder first computes ML-estimate using SD, and then generates a list of candidate vectors around ML solution during a second SD run. In [152] a VLSI implementation of a soft-output LSD has been presented and shown that LSD implementation is efficient for small list size, where as large list size leads to high computational complexity. In [153], two kinds of iterative sphere decoders based on modification of the Schnorr-Euchner enumeration were presented. Other recent implementations based on soft-input soft-output sphere decoder were described in [154, 155]. In [156, 134], an extension of LSD by tuple search algorithm was introduced in order to further reduce the detection complexity through the use of two separate lists: a candidate list for LLR computation and a search tuple for the search space.

Missing counter hypothesis problem

In a list based detection, the key challenge is to ensure that the list contains MAP estimate and counter hypotheses. However some counter hypotheses are missing, hence LLR values could not be determinate. To address this problem, several techniques have been proposed as briefly presented in this section.

LLR clipping: This method is very simple for implementation, it consists in setting the LLR to a predefined or fixed value. LLR clipping level has a significant impact on the system performance. This level depends on system configuration, especially on the SNR, on the list size and on the modulation order. It is obvious that the probability of missing counter hypotheses increases as the list size decreases and the modulation order increases. For a large list size, LLR clipping level must be chosen relatively high; $L_{clip} = 8$ was used in [27, 101] and was given good results. While in small to moderate list size, a low clipping level must be used ($L_{clip} = 3$). In [157, 158], a SNR-aware approach for calculating LLR clipping level was considered. A multilevel clipping

level was then proposed in [159]. The optimal LLR clipping level must be therefore determined for a given system based on maximizing the average mutual information at the detector output.

Bit flipping/Chase decoding: To compute the LLR of a bit, the counter hypothesis is generated by flipping the corresponding bit of MAP candidate and computing the Euclidean distance [160, 161, 162]. In this method, LLR value does not have a sufficient reliability due to the coupling between layers [148]. In addition, more computational complexity is required.

Last list entry: In this technique, the worse Euclidean distance of the list is used as a lower bound of the missing counter hypothesis [163]. LLRs are then computed based on MAP distance and on this worse case distance, resulting on a very aggressive clipping value. This method leads to a significant performance loss especially in the case of small list size.

Path augmentation/Candidate adding: Path augmentation method was used in the list sequential decoder to extend the incomplete path in the stack algorithm using either a linear estimation or a soft *a priori* information [95, 164]. This method requires an additional complexity especially with a large number of extended paths.

As a conclusion, clipping method seen to be more suitable for implementation with a negligible computational complexity. However, an appropriate clipping level must be selected for a good system performance.

3.5.2 Single tree-search sphere decoder (STS-SD)

One of the two minima in equation (3.4) corresponds to MAP hypothesis $\mathbf{s}^{MAP}$, while the other corresponds to the counter hypothesis. The computation of LLR can be done as:

$$L\left(x_{i,b}\right) = \begin{cases} \frac{1}{N_0}\left(d_{i,b}^{\overline{\mathrm{MAP}}} - d^{\mathrm{MAP}}\right), & \text{if } x_{i,b}^{\mathrm{MAP}} = +1 \\ \frac{1}{N_0}\left(d^{\mathrm{MAP}} - d_{i,b}^{\overline{\mathrm{MAP}}}\right), & \text{if } x_{i,b}^{\mathrm{MAP}} = -1. \end{cases}$$

$$d^{\mathrm{MAP}} = \left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}^{\mathrm{MAP}}\right\|^2 - N_0 \log P\left(\mathbf{s}^{\mathrm{MAP}}\right), \tag{3.39}$$

$$d_{i,b}^{\overline{\mathrm{MAP}}} = \min_{s \in \chi_{i,b}^{\overline{\mathrm{MAP}}}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 \log P\left(\mathbf{s}\right)\right\}, \tag{3.40}$$

$$\mathbf{s}^{\mathrm{MAP}} = \arg \min_{s \in 2^{Q \cdot N_t}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 \log P\left(\mathbf{s}\right)\right\}, \tag{3.41}$$

where $\chi_{i,b}^{\overline{\mathrm{MAP}}}$ denotes the bit-wise counter hypothesis of the MAP hypothesis, which is obtained by searching over all the solutions with the b^{th} bit of the i^{th} symbol opposite to the current MAP hypothesis.

Originally, MAP hypothesis and counter hypotheses can be found through repeating the tree search [160] that requires a large computational complexity cost. This algorithm denoted as repeated tree search (RTS) transforms the decoding problem into set of hard SD problems. Therefore, one hard SD step is used to find $\mathbf{s}^{MAP}$, then N_t SD steps are used to find counter hypotheses for each bit. Hence, RTS entails a large number of redundant computations.

To overcome this, a parallel implementation of soft-output sphere decoder has been developed to allow simultaneous computation of LLR values [91]. This technique is later denoted as single tree search (STS) in [133]. A VLSI implementation of soft-output STS-SD has been presented using sorted QR-decomposition, channel matrix regularization and LLR clipping. The soft-output STS-SD was then extended to soft-input soft-output SD for iterative MIMO receiver in [28]. This algorithm incorporates the clipping of extrinsic LLRs into the tree-search, which results in significant complexity reduction.

The basic idea of STS-SD is to search the sub-tree originating from a given node if the Euclidean distance leads to an update of either d^{MAP} or at least one of $d_{i,b}^{\overline{\mathrm{MAP}}}$ using an administration list and an appropriate pruning criterion. The algorithm maintains an administration list that contains MAP hypothesis, its corresponding metric and all QN_t extrinsic metrics. The algorithm is initialized with $d^{\mathrm{MAP}} = d_{i,b}^{\overline{\mathrm{MAP}}} = \infty$ and $x_{i,b}^{MAP} = 1$. Whenever a leaf node, x, is reached, two cases are considered:

1. MAP hypothesis update: if $d(x) < d^{\mathrm{MAP}}$, a new MAP hypothesis is found. All extrinsic metrics are updated according to $d_{i,b}^{\overline{\mathrm{MAP}}} = d^{\mathrm{MAP}}$, followed by updating $d^{\mathrm{MAP}} = d(x)$.
2. Extrinsic metric update: if $d(x) > d^{\mathrm{MAP}}$, only extrinsic metric must be updated if $d(x) < d_{i,b}^{\overline{\mathrm{MAP}}}$.

Channel matrix regularization and runtime constraints may be also used in STS-SD to reduce the decoding complexity and to meet the practically requirements for a fixed throughput at the price of performance degradation [79, 165]. A straightforward way to set the runtime constraints is to terminate the search after a maximum number of visited nodes. The compensation of self-interference using MMSE-SQRD can be incorporated directly into the tree search procedure to recover the performance loss.

Through the use of extrinsic LLR clipping method, STS-SD algorithm can be tunable between MAP performance and hard-output performance. The implementations of STS-SD have been reported in [137, 138].

3.5.3 SISO K-Best decoder

K-Best algorithm based on breadth-first search has been also modified to support soft-output information. As the list sphere decoder, this algorithm performs a tree search to find a list of K best candidates. LLRs are then computed from this list at the end of the algorithm. We note that the candidate list does not necessarily correspond to lowest Euclidean distances.

In [101], an iterative tree search detection based on M-algorithm was proposed. In [29] a hard-output K-Best Schnorr Euchner (KSE) decoder and a soft-output MKSE decoder have been proposed and implemented for a 16-QAM 4×4 MIMO system. The soft-outputs are generated by using the discarded paths in addition to the K survivor paths. The results show that the proposed algorithms and the VLSI architecture are feasible to achieve near-ML performance with high throughput detection and reasonable complexity. A new detection technique based on M-algorithm was proposed in [162] by taking into account the influence of unvisited paths on computing path metrics in each level for the K best candidates using a bias term. In [166], an improved K-Best

algorithm was proposed by running K-Best search several times to visit candidates inside the radius but outside the list.

The main drawbacks of K-Best decoder are the expansion and the sorting operations as discussed in section (2.4.3). Moreover, K-Best decoder suffers similarly as LSD from missing counter hypothesis problem due to the limited list size. Numerous approaches have been proposed to address this problem such as smart candidates adding [167], bit flipping [162], path augmentation and LLR clipping [27, 101] as discussed in section (3.5.1).

3.5.4 SISO Fixed sphere decoder

Fixed sphere decoder algorithm is extended to support soft-input soft-output detection for the iterative MIMO receiver [135, 168]. In the soft-output FSD proposed in [135], the search is performed not only to find ML solution but also to find a set of candidates around ML solution in order to compute the LLR of all bits. Therefore, a subset S is first chosen, then ML solution of the subset is used to generate a subset S'. The combined list $\{S \cup S'\}$ is finally used to compute an approximation of the extrinsic LLR.
In [136], a first VLSI architecture of SISO FSD was presented with four nodes per cycles parallel architecture. Efficient SISO FSD implementations have been reported in [141, 142].

3.6 Lattice reduction aided detection

We have previously seen in section (2.5), that lattice reduction methods can be combined with other detection algorithms in order to improve their performance. The detection is therefore performed in a reduced domain. In order to provide soft information, a set of candidates must be generated in the reduced domain and transformed back into the original signal space as presented in [169, 170]. This list generated in the reduced domain can be considered as an additional neighborhood. This neighborhood is actually seen as a set of displacements $\mathbf{D}$ commonly known as a perturbation matrix. Despite of the low computational complexity of perturbation method, its performance remains very sub-optimal since it does not able to offer a list in the original domain with highest probability of transmitted symbol candidates.

Nearest neighbor is another method proposed in order to generate a list though modifying the distance criterion [170]. This solution presents an improvement compared to perturbation method. However, it still suffers from a large SNR loss compared to optimal detection especially with high-order modulation. Other approaches based on sphere decoding neighborhood generation were described in [171].

The main disadvantages of LR-aided detection are the high preprocessing step required to compute the channel matrix H using LLL algorithms and the additional neighborhood set require to generate the LLRs. The reader can refer to [169, 170, 171] for more details. This approach will not be discussed in the remainder of this thesis.

3.7 Low-complexity K-Best (LC-K-Best) decoder

Classical K-Best decoder computes $K\sqrt{2^Q}$ Euclidean distances. Then, a sorting operation is done to choose the K best candidates as illustrated in Figure 3.4 with an example of $K = 4$. The proposed K-Best decoder denoted as LC-K-Best decoder in the sequel, uses two improvements over the classical K-Best decoder for the sake of lower complexity and latency [30].

Simplified hybrid enumeration

The first improvement simplifies the hybrid enumeration of constellation points in real system model when the *a priori* information is incorporated into the tree search using two look-up-tables (LUTs). Hybrid enumeration was initially proposed in [172] for soft-input sphere decoder in complex system model to avoid exhaustive calculation and sorting of partial metrics of constellation points required in SE enumeration. It consists in separating the partial metric into two metrics: the channel metric and the *a priori* metric. The proposal is to simplify the enumeration in the real-valued system model through the use of two look-up tables (LUT) in order to reduce the computational complexity of the receiver. The first LUT is used to find the entry for SE enumeration based on the channel metric m_i^C. The second LUT is filled with the *a priori* metric m_i^A in ascending order. The enumeration is approximated through the orthogonality of these two metrics. Figure 3.5 illustrates an example of the enumeration strategy. First, the constellation points are enumerated according to m^C and m^A and stored in the LUTs. Then, the smallest Euclidean distances of m^C and m^A are compared (S2 and S3). The one which has the minimum distance (S2 in m^C) is chosen as the first point. Then, the first point in m^A (S3) is compared to the next point in m^C (S1). Since S3 has a lower distance, it is considered as the second point and so on.

Relaxed on-demand expansion

The second improvement is to use a relaxed on-demand expansion that reduces the need of exhaustive expansion and sorting operations. The on-demand expansion was proposed in [98] for hard-output decoder. It consists in expanding the first children of parent nodes and choosing one minimum between these children. Then, the survival path expands the next child. This operation visits $2K - 1$ nodes at each level to get K best nodes. In our approach, a portion A of the first children is chosen. Then, the corresponding parents expand their next children. This operation is repeated to get K best nodes. The number of the first children A is chosen in order to allow parent node to extend all its possible children nodes depending on the constellation and on the total number K of retained solutions. Therefore the number of visited nodes will be reduced to $2K - A$ at each level. Figure 3.4 shows an example with $K = 4$ and $A = 2$. All parent nodes at the first level expand their first children. The two children that have the smaller Euclidean distances (nodes 1 and 7) are retained. Then, the corresponding parent nodes (P1 and P2) expand their next children (nodes 3 and 8). The distances are compared, and the two nodes (3 and 10) having the lowest distances are retained to get 4 best candidates.

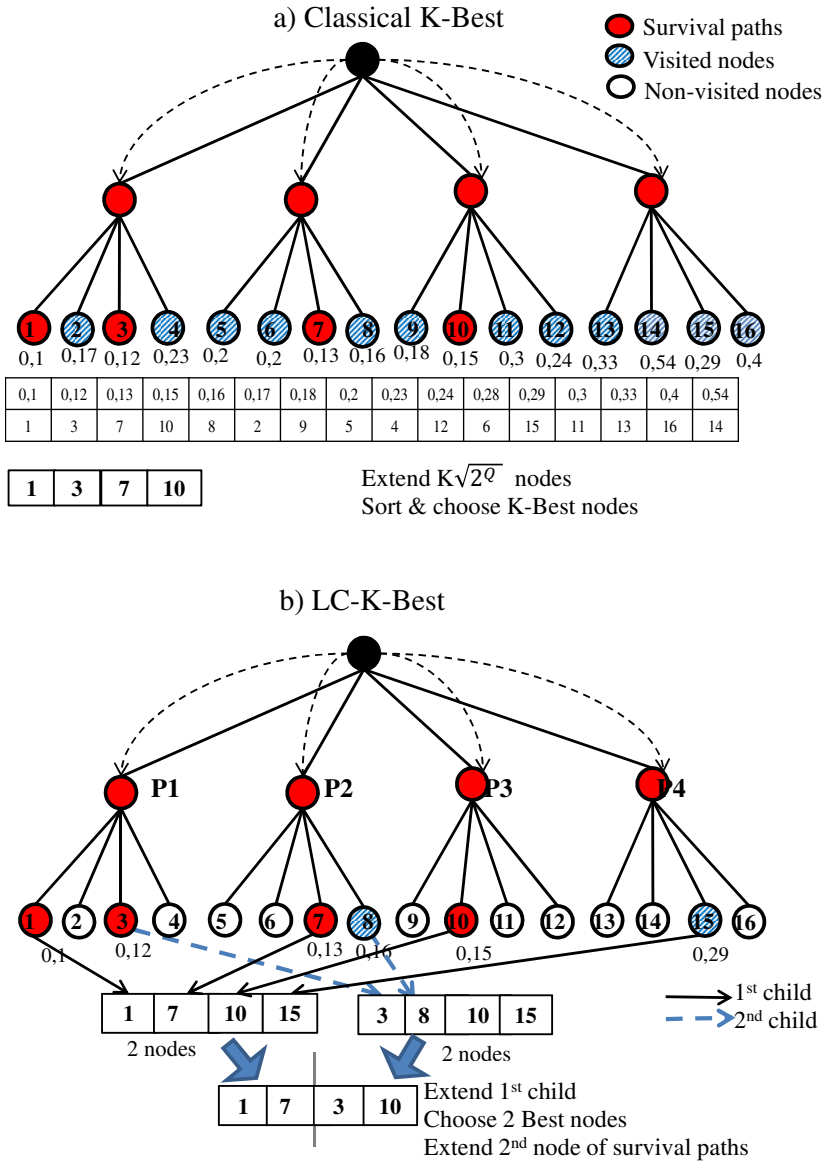


Figure 3.4: Classical K-Best versus LC-K-Best, $K = 4$, **(a)** Classical K-Best, and **(b)** LC-K-Best.

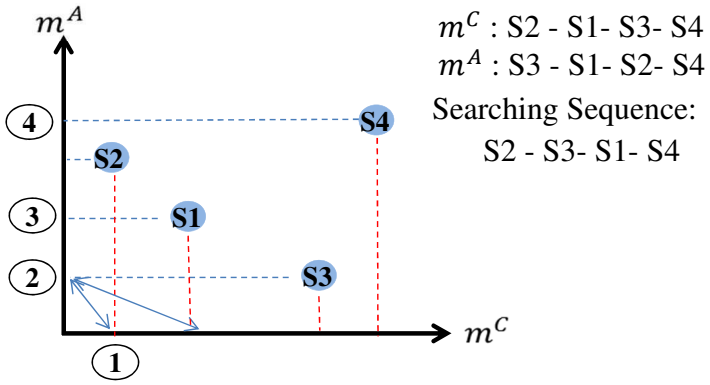


Figure 3.5: Enumeration strategy based on $\boldsymbol{m^C}$ and $\boldsymbol{m^A}$.

LC-K-Best algorithm can be described as follow:
Preprocessing step:

1. Input: $\mathbf{y}$,$\mathbf{H}$, K
   Calculate: $\mathbf{H} = \mathbf{QR}$, $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$
2. Enumerate the constellation symbols based on m^A for all layers

Tree-search step:

1. Set layer to $2N_t$, list $\mathcal{L} = \emptyset$, $d_{2N_t+1} = 0$
   (a) Expand all $\sqrt{(2^Q)}$ possible constellation nodes
   (b) Calculate their corresponding PEDs
   (c) if $\sqrt{(2^Q)} > K$, select the K best nodes and store them in the list $\mathcal{L}_{2N_t}$,
2. For layer $i = 2N_t - 1 : 1$
   (a) Enumerate the constellation point according to m_i^C of K surviving paths in the list $\mathcal{L}_{i+1}$,
   (b) Find the first child (FC) based on m_i^C and m_i^A for each parent nodes,
   (c) Compute their PEDs,
   (d) Select the A best children having the smallest PEDs among the K FCs and add them to the list $\mathcal{L}_i$,
   (e) if $|\mathcal{L}_i| < K$, Find the next child (NC) of the selected parent nodes, Calculate their PEDs and go to step 2.d.
   (f) else move to the next layer $i = i - 1$ and go to step 2.
3. if $i = 1$, Calculate the LLR as in equation (3.38).

Table 3.3 illustrates the reduction in the number of visited nodes for different modulation orders. With our novel search strategy, the computational complexity and decoding throughput can be significantly improved specially in case of high modulation order. For instance, a reduction of about 50% is observed in case of 4×4 16-QAM, and the reduction can be as high as 75% for 64-QAM case.

Table 3.3: Reduction ratio of the number of visited nodes in the tree-search with different modulation order.

	K-Best	LC-K-Best	Complexity reduction
4×4 4-QAM $K = 8$	94	74	21%
4×4 16-QAM $K = 16$	404	188	53%
4×4 64-QAM $K = 64$	3144	792	75%

Figure 3.6 shows the performance of LC-K-Best decoder compared to classical K-Best decoder for different constellation sizes with $I_{in} = 8$ inner iterations and $I_{out} = 1$, 4 outer iterations. STS-SD is plotted in dotted line as a reference since it offers optimal performance. We can see that LC-K-Best algorithm has almost the same BER performance compared to classical K-Best for all constellations and iterations. We show also that the performance of the system is improved through iterations by about 1 dB in the case of 4-QAM and 1.5 dB in case of 16-QAM and 64-QAM at a BER level

of 10^{-5}. Additionally, at the first iteration, STS-SD outperforms the proposed K-Best decoder by about 0.5 dB with 16-QAM and 0.25 dB with 4-QAM and 64-QAM at a BER level of 1×10^{-4}. However, with 4 outer iterations, this gap is reduced to 0.2 dB with 16-QAM and 0.1 dB with 4-QAM and 64-QAM.

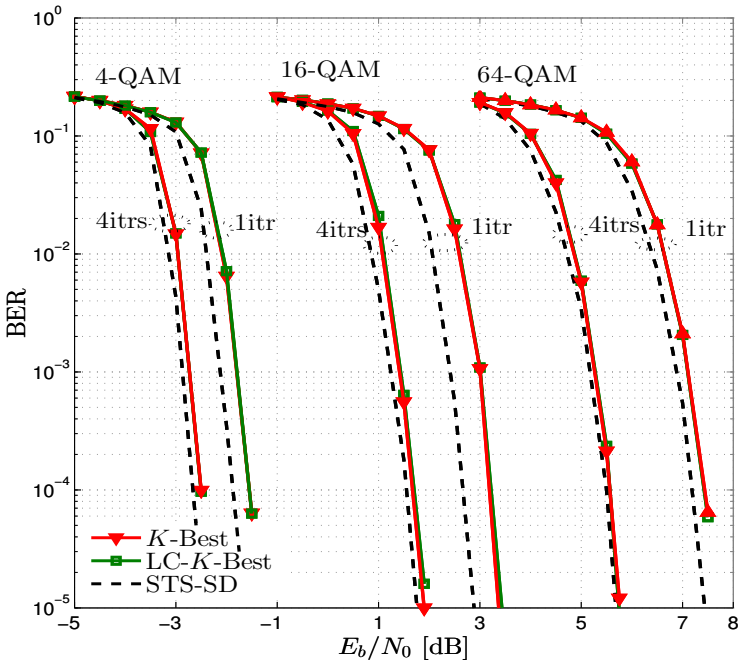


Figure 3.6: Comparison of BER performance for K-Best and LC-K-Best at different E_b/N_0 values with $I_{in} = 8$ turbo decoding iterations and $I_{out} = 1, 4$ outer iterations in a $\mathbf{4 \times 4}$ coded MIMO system using 4-QAM, 16-QAM and 64-QAM, R_c =1/2, $K_b = 2,048$.

The performance-complexity tradeoff achieved by LC-K-Best is interesting. Therefore this approximation will be considered in our convergence analysis and performance evaluation in the following sections.

3.8 Convergence analysis using EXIT charts

3.8.1 EXIT chart principle

Extrinsic Information Transfer (EXIT) chart proposed in [173] is an useful tool to study the convergence behavior of iterative decoding systems. In the following, we briefly describe the concept of EXIT chart and explain its construction methodology.

EXIT chart is an analytical technique to describe the exchange of mutual information across the iterative process in order to predict the required number of iterations, the convergence threshold (corresponding to the start of the waterfall region) and the average decoding trajectory.

To explain the construction methodology, we consider as an example a serial concatenation of two soft-input soft-output decoders as illustrated in Figure 3.7. The inner decoder takes the channel observations $\mathbf{y}$ and *a priori* LLRs L_{A1} and outputs extrinsic LLRs L_{E1} which are then passed through the bit interleaver, to become *a priori* information L_{A2} for the outer decoder. The latter feeds back in turn extrinsic LLRs L_{E2},

which are reinterleaved and become new *a priori* information for the inner decoder. Let I_{A1} and I_{A2} denote the *a priori* mutual input information of the inner and outer decoders respectively. I_{E1} and I_{E2} denote their corresponding extrinsic mutual output information.

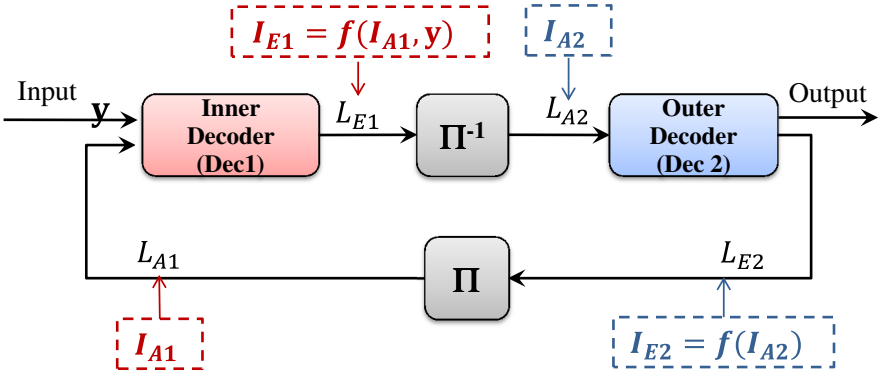


Figure 3.7: Block diagram of 2 serially concatenated decoders.

The inner and the outer decoders can be described by the extrinsic information transfer characteristics. These characteristics correspond to the transfer functions of the decoders where $I_{E1} = f(I_{A1}, \mathbf{y})$ and $I_{E2} = f(I_{A2})$. We note that the outer decoder only depends on the mutual information at its input since it is not connected to the channel. EXIT chart is obtained by superposing these two characteristics in the same graph.

At the beginning, the *a priori* mutual information $I_{A1} = 0$ and $I_{A2} = 0$. Then, the extrinsic mutual information I_{E1} of the inner decoder (Dec1) becomes the *a priori* mutual information I_{A2} of the outer decoder (Dec2) and vice versa (i.e. $I_{E1} = I_{A2}$ and $I_{E2} = I_{A1}$) as illustrated in Figure 3.8. For Dec1, I_{A1} and I_{E1} are placed on the abscissa and the ordinate of the graph, respectively. However for Dec2, the axes are swapped.

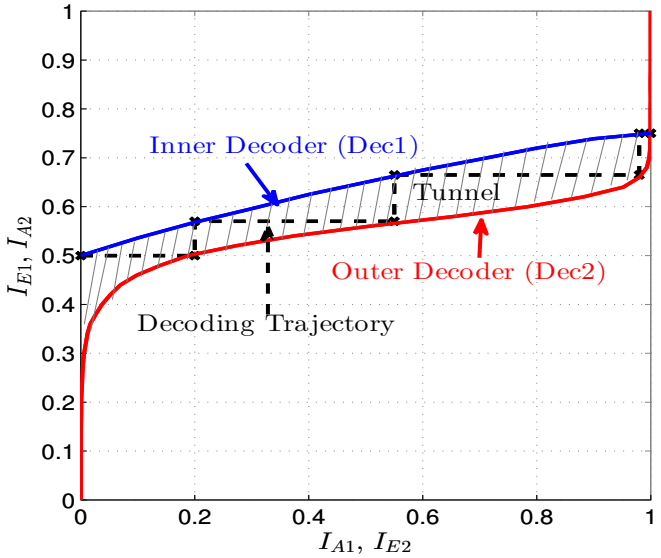


Figure 3.8: EXIT chart of two concatenated decoders (Dec1 and Dec2) and their corresponding decoding trajectory.

For a successful decoding, there must be an open tunnel between the curves. The exchange of extrinsic information can be visualized as a *zig-zag* decoding trajectory in EXIT chart as shown in Figure 3.8. Jumping from one curve to the other to reach a mutual information near from one determines the number of iterations. Crossing for a mutual information of one corresponds to the convergence point; meaning that if the *a priori* information is entirely correlated with the initial information, then the decoder will necessary output an extrinsic information entirely correlated with the initial information. The convergence threshold corresponds to minimum value of SNR beyond which successive iterations begin to bring improvements in BER (for which we observe a clear path between the curves).

By assuming that the transmitted coded bits (x) are i.i.d, the mutual information I_x (I_A or I_E) can be computed by means of Monte Carlo simulation using the probability density function p_{L_x}:

$$I_x = \frac{1}{2} \sum_{x \in \{-1,1\}} \int_{-\infty}^{+\infty} p_{L_x}(L_x|x) \log_2 \frac{2p_{L_x}(L_x|x)}{p_{L_x}(L_x|-1)\, p_{L_x}(L_x|+1)} dL_x. \tag{3.42}$$

Under the assumption of large interleaving and by applying the consistency condition, I_x can be approximated by [174]:

$$I_x \approx 1 - \frac{1}{L_b} \sum_{n=0}^{L_b-1} \log_2 \left(1 + \exp\left(-xL_x\right)\right), \tag{3.43}$$

where L_b is the number of transmitted bits, L_x is the LLR associated with the bit $x \in \{-1, +1\}$.

The *a priori* information L_A can be modeled by applying an independent Gaussian random variable n_A with zero mean and variance σ_A^2 in conjunction with the known transmitted information bits x [31]:

$$L_A = \mu_A x + n_A, \quad \text{where} \quad \mu_A = \sigma_A^2/2. \tag{3.44}$$

The mutual information I_A can be then expressed in function of σ_A, $I_A = J\{\sigma_A\}$.
To trace EXIT chart, we generate for each decoder the *a priori* information according to a given mutual information at its input (I_A between 0 and 1), then we determine the mutual information of the extrinsic information at the output. A large interleaver is required for an accurate EXIT chart.

In the iterative receiver considered in our study, two iterative processes are performed, one inside the channel decoder and the other between the MIMO detector and the channel decoder. Hence, an 3D EXIT chart can be used to visualized the convergence behavior of the overall system. For simplicity, we separately study the convergence of the channel decoding and the MIMO detection as illustrated in Figure 3.9. The MIMO detector and the channel decoder exchange information about coded bits (systematic and parity bits). Meanwhile in the channel decoder, only *a priori* information about systematic bits is exchanged. We denote by I_{A1} and I_{A2} the *a priori* mutual input information of the MIMO detector and the channel decoder respectively. I_{E1} and I_{E2} denote their corresponding extrinsic mutual output information. For the SISO components of the channel decoder, the exchange of mutual information

is expressed by I_{Ai1}, I_{Ai2},I_{Ei1} and I_{Ei2}. In a first time, the convergence inside the turbo decoder is considered. Next, we analyze the convergence of the channel decoder as one block and the MIMO detector.

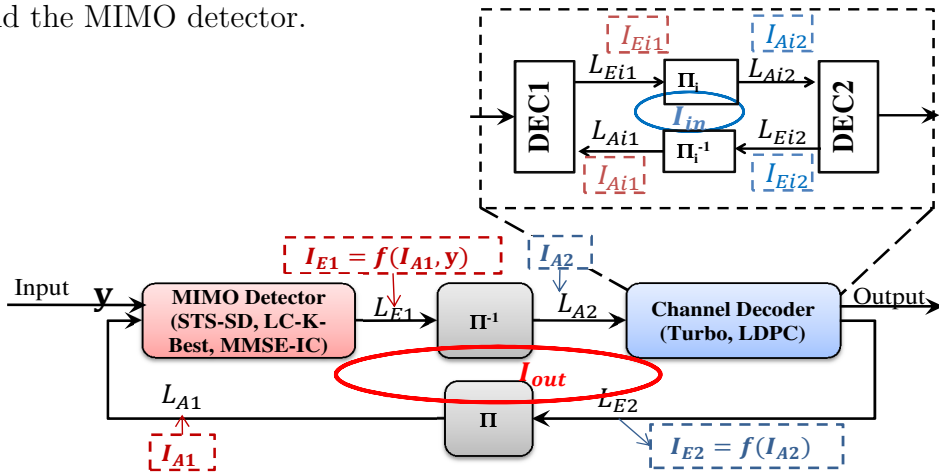
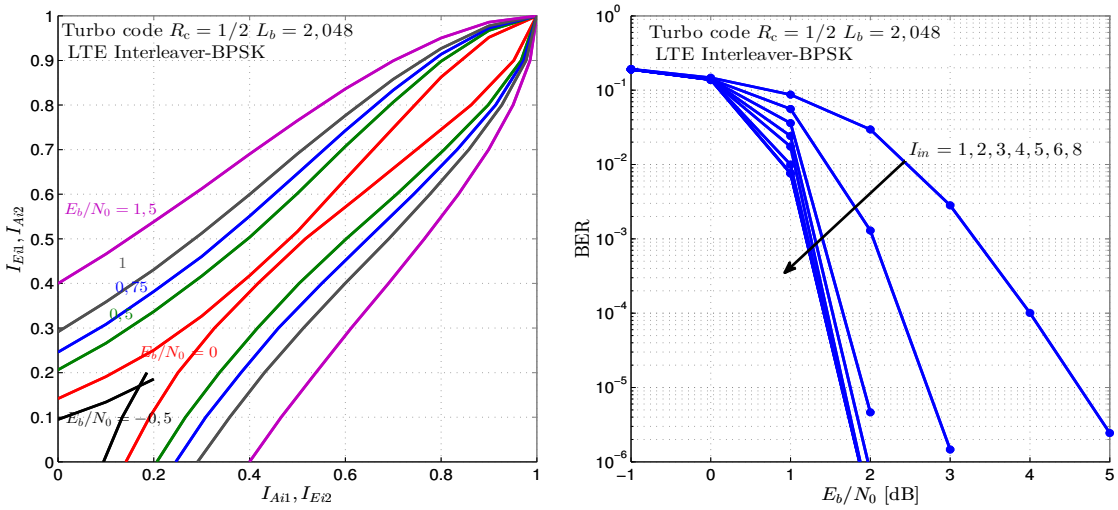


Figure 3.9: EXIT chart block diagram for iterative MIMO detection and decoding.

3.8.2 Convergence of turbo decoder

The convergence of turbo decoder has been widely investigated in the literature. The impact of memory size, polynomial generators as well as different component decoders on the convergence behavior of turbo decoder has been studied in [31].

Figure 3.10a shows EXIT chart of the two component decoders of LTE turbo decoder for different E_b/N_0 values with BPSK modulation in AWGN channel. We show that at low E_b/N_0 (-0.5 dB), the two characteristics intersect. By increasing E_b/N_0 value, the transfer characteristics are shifted upward, an open tunnel can be clearly observed at $E_b/N_0 = 0$ dB, allowing the iterative process to bring improvement to the system. This E_b/N_0 value corresponds to the turbo cliff position in BER curve as shown in Figure 3.10b.



(a) EXIT chart **(b)** BER performance

Figure 3.10: EXIT chart of the two component decoders of LTE turbo decoder **(a)**, and BER performance **(b)**, $R_c = 1/2$, $K_b = 2,048$ in AWGN channel with BPSK modulation for different E_b/N_0 values.

In Figure 3.11, the iterative decoding trajectory for different E_b/N_0 values is drawn and it approximately matches the corresponding EXIT chart. We show that at $E_b/N_0 = 0$, more than 8 iterations are required. However less iterations are required with the increase of E_b/N_0 values. Increasing the number of iterations above 6 and 4 in the case of $E_b/N_0 = 0.75$ dB and $E_b/N_0 = 1$ dB, respectively does not bring significant improvement on the system performance.

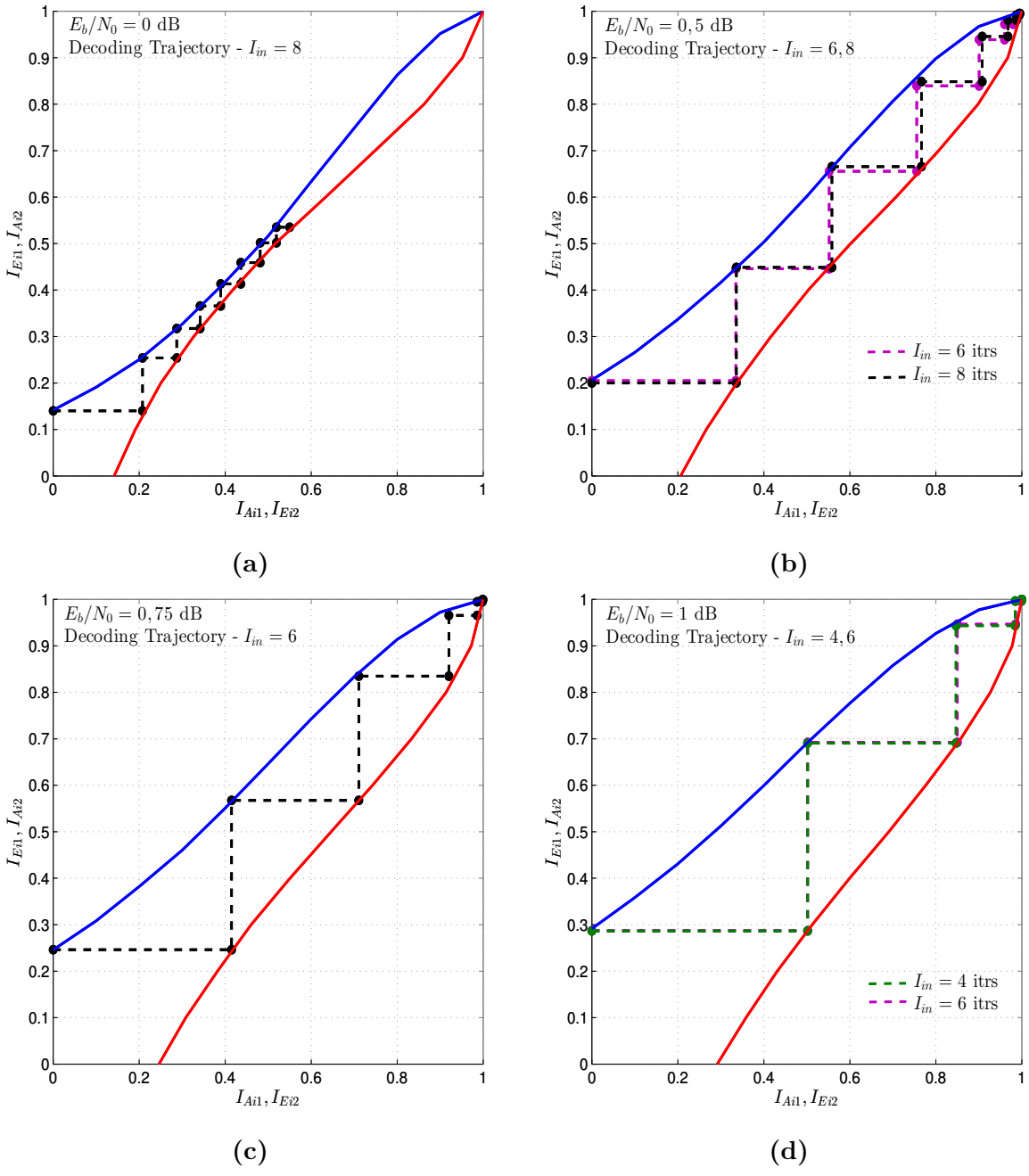


Figure 3.11: Iterative decoding trajectory of LTE turbo decoder, $R_c = 1/2$, $K_b = 2,048$ in AWGN channel with BPSK modulation for different E_b/N_0 values.

Figure 3.12 shows the extrinsic information transfer characteristics of LTE turbo decoder with $R_c = 1/2$, $K_b = 2,048$ and $I_{in} = 1$ to 8. Note that the axes are swapped, the input is on the ordinate and the output is on the abscissa. We can see that after six to eight iterations, there is no further improvement on the mutual information.

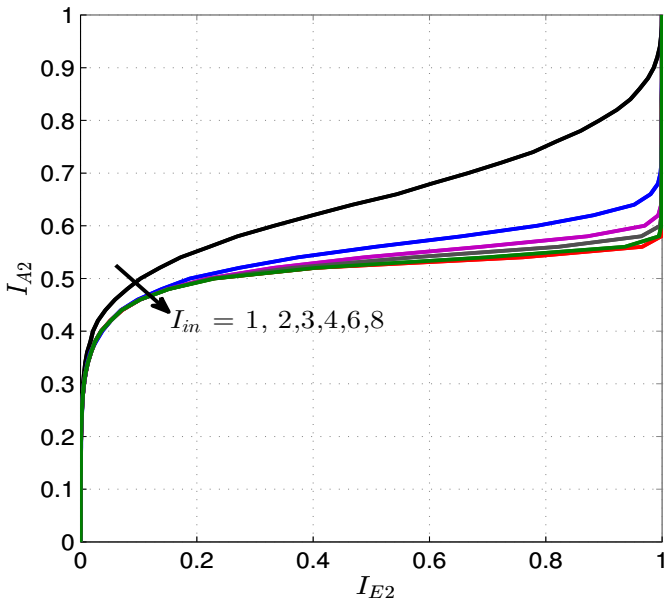


Figure 3.12: Extrinsic information transfer characteristics of LTE turbo decoder, with $I_{in} = 1, 8$, $R_c = 1/2$, $K_b = 2,048$.

3.8.3 Convergence of LDPC decoder

LDPC decoder requires many iterations to successively decode the received block. In the literature, typically 20 to 50 iterations are performed within the LDPC decoder to get near-optimal performance. The convergence behavior of LDPC decoder can be studied by considering the check node (CN) and variable node (VN) component decoders (cf. section 1.3.2.2).

In [175], LDPC code is optimized by performing a curve fitting on extrinsic information transfer charts in order to design the code for iterative decoding, i.e., how to choose good degree distributions for the modulator, the channel, and the detector. More analysis of LDPC codes based on EXIT charts was presented in [176], in which exact expressions for EXIT functions of repetition and single-parity check codes were proposed.

Figure 3.13 shows the extrinsic information transfer characteristics of LDPC decoder with $R_c = 1/2$, $N_b = 1,944$ and I_{in} ranging from 1 to 50. It is obvious that the extrinsic mutual information increases with the number of iterations. We show that 20 iterations are enough for LDPC decoder to converge. Increasing this number to 50 iterations does not bring significant improvement on the mutual information at its output.

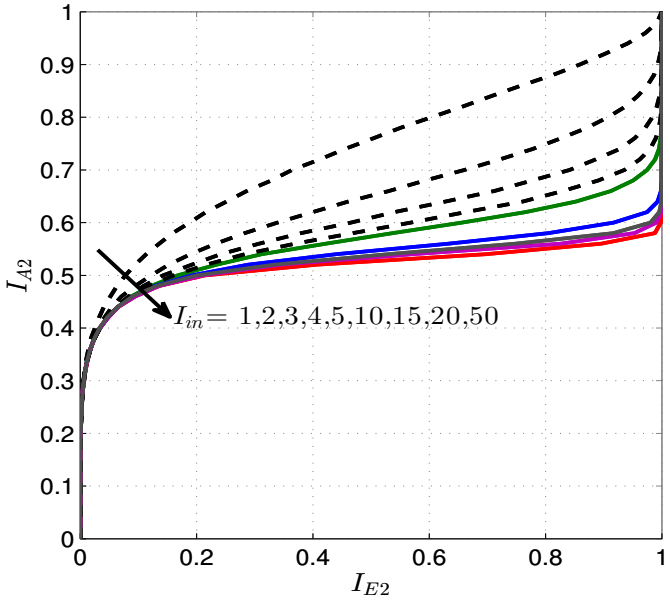


Figure 3.13: Extrinsic information transfer characteristics of LDPC decoder, with $I_{in} = 1, 50$, $R_c = 1/2$, $N_b = 1,944$.

3.8.4 Convergence behavior of the iterative receiver

To visualize the exchange of extrinsic information of the iterative receiver, we plot the detector and the decoder characteristics into a single chart. For our convergence analysis, a 4×4 MIMO system with 16-QAM constellation, turbo decoder and LDPC decoder ($R_c = 1/2$) is considered. In the first time, a detailed convergence analysis of MIMO detection with LTE turbo decoder is investigated. We then give a brief analysis of the convergence of MIMO detection with LDPC decoding. Table 3.4 summarizes the principle parameters for the convergence analysis.

Table 3.4: Simulation parameters.

MIMO system	4×4 Spatial multiplexing
Channel type	Flat Rayleigh fading
Modulation 2^Q-QAM	16-QAM with Gray mapping
Detector	STS-SD LC-K-Best $K = 16$ MMSE-IC
Channel decoder	LTE turbo code $(13, 15)_o$ Block Length $K_b = 2,048$, $R_c = 1/2$
	LDPC code (IEEE 802.11n) Codeword Length $N_b = 1,944$, $R_c = 1/2$
Interleaver	Random, size $= 2,048$ (turbo code case)

Convergence of MIMO detection with LTE turbo decoding

Figure 3.14 shows EXIT chart of turbo decoder by applying different outer iterations at $E_b/N_0 = 1.5$ dB. The red curves correspond to the non iterative detection decoding while other colored curves correspond to several iterations. We can see that with no iteration (red curve) the tunnel is blocked, convergence can not be attained. Applying more iterations enlarges the tunnel. For STS algorithm, 4 outer iterations are sufficient to LTE turbo decoder to converge. In the case of LC-K-Best decoder, the tunnel will be also enlarged for each outer iteration. However the tunnel is wider in the case of STS decoder, more than 4 iterations are required inside the turbo decoder to reach a mutual information of one. The same result is observed in the case of MMSE-IC which presents a lower mutual information than LC-K-Best decoder. The exchange of extrinsic information can be visualized as a *zig-zag* decoding trajectory in the EXIT chart. With 4 iterations, we notice that the trajectory closely matches the characteristics in case of STS algorithm, while for the LC-K-Best and MMSE-IC algorithms, the decoding trajectory slightly diverges from the characteristics towards smaller extrinsic output after few iterations.

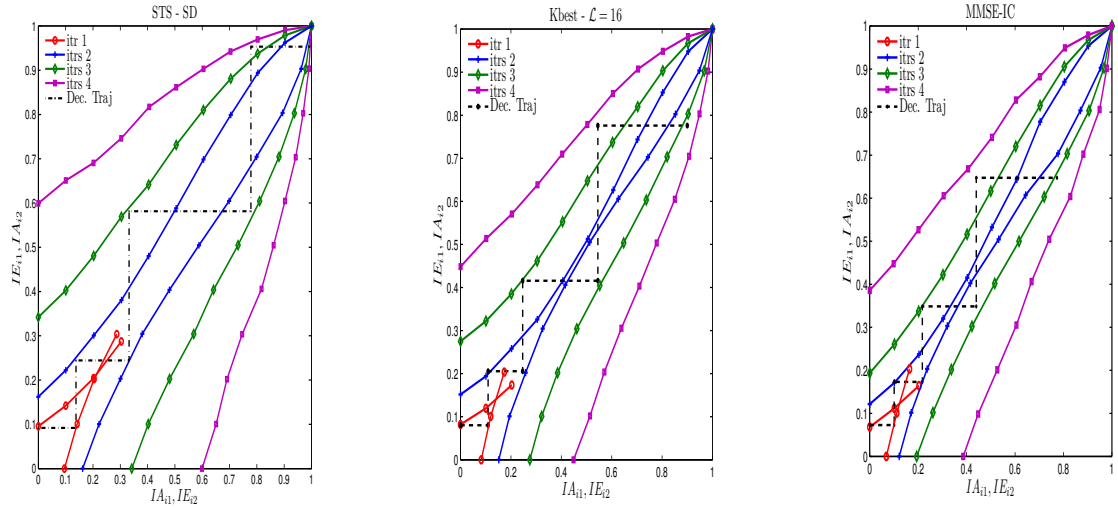


Figure 3.14: EXIT chart of the two component decoders inside LTE turbo decoder ($R_c = 1/2$, $I_{in} = 1$) for $I_{out} = 1 - 4$ with several detectors (STS-SD, LC-K-Best, MMMSE-IC) at $E_b/N_0 = 1.5$ dB in a $\mathbf{4 \times 4}$ MIMO system using 16-QAM.

Figure 3.15 shows EXIT chart of the overall system for different E_b/N_0 values and several MIMO detectors (STS-SD, LC-K-Best, MMSE-IC and MMSE-IC1). As I-VBLAST detector performs successive interference cancellation at the first iteration, and parallel interference cancellation of the soft estimated symbols for the rest iterations, it is less intuitive to present its convergence in the EXIT chart. Therefore, the convergence analysis of VBLAST is not given in this work.

We notice in Figure 3.15 that the characteristics of turbo decoder is independent of E_b/N_0 values. At a low signal to noise ratio (e.g, $E_b/N_0 = 1$ dB), with 1 inner iteration, MIMO detector and turbo decoder transfer characteristics intersect at low mutual information (≤ 0.4), the tunnel is blocked for all detection algorithms. Therefore, the

performance cannot be improved through iterations resulting in high BER. With the increase of E_b/N_0, the transfer characteristics of MIMO detectors are shifted upward, the tunnel between the two characteristics is then open (at about 1.5 dB) allowing the iterative process to improve the performance of the system. This E_b/N_0 value corresponds to the turbo cliff position as verified in the BER performance. For high E_b/N_0 values, less iterations are required to reach low BER.

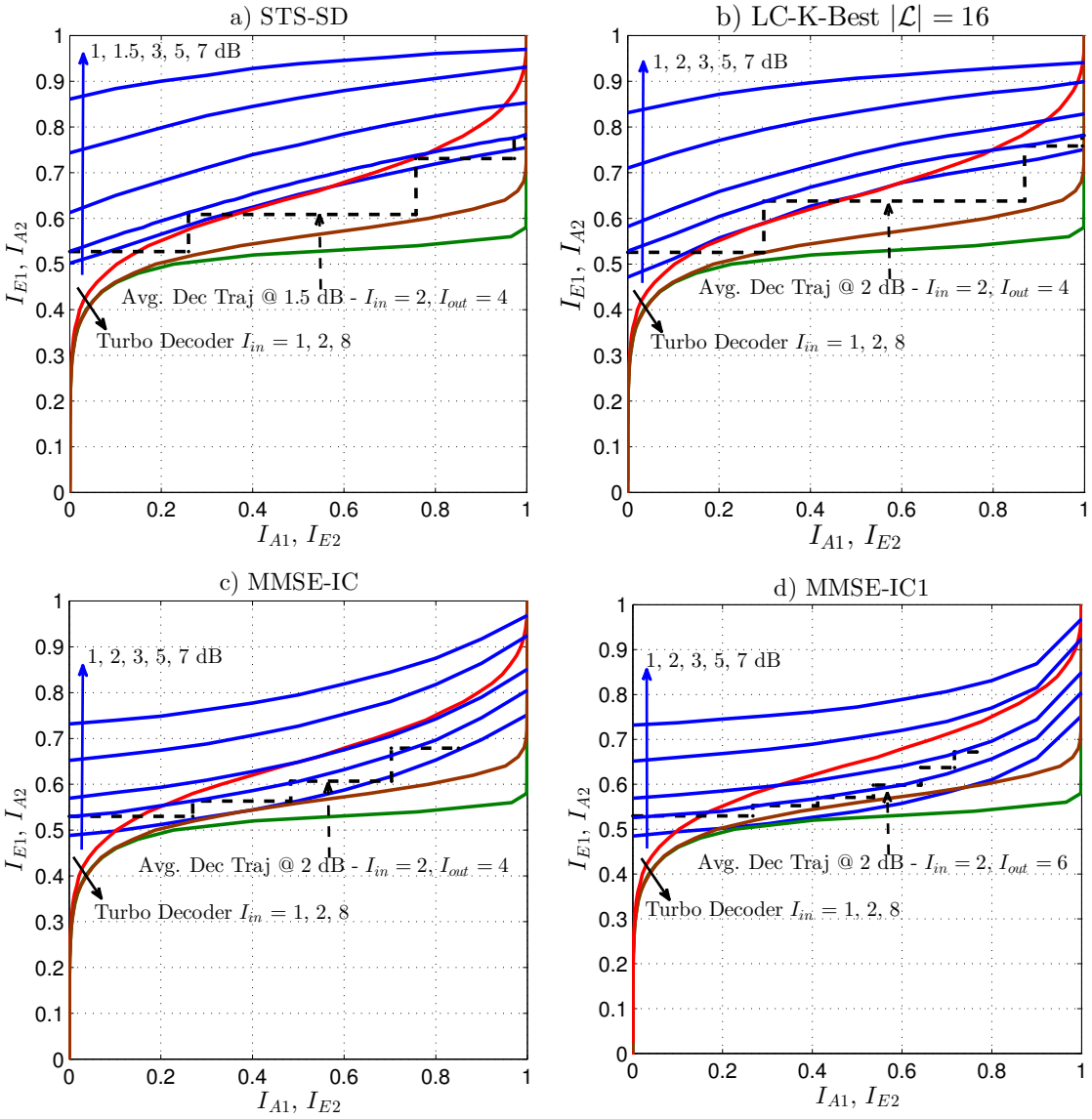


Figure 3.15: EXIT chart of LTE turbo decoder ($R_c = 1/2$) and MIMO detectors: **(a)** STS-SD, **(b)** LC-K-Best, **(c)** MMSE-IC, **(d)** MMSE-IC1, at different E_b/N_0 values (1, 1.5, 2, 3, 5, 7 dB) in a $\mathbf{4 \times 4}$ MIMO system using 16-QAM.

By comparing the characteristics of STS-SD, LC-K-Best decoder and MMSE-IC equalizers, we notice that STS-SD has a larger mutual information at its output. LC-K-Best decoder has slightly less mutual information than STS-SD. MMSE-IC and MMSE-IC1 show low mutual information levels at their outputs compared to other algorithms when $I_{A1} < 0.85$, while for $I_{A1} > 0.85$, the extrinsic mutual information is similar to others.

Furthermore, as shown in Figure 3.15a, in the case of STS-SD, at $E_b/N_0 = 1$ dB with 8 iterations inside the turbo decoder, the tunnel is open and hence applying outer iterations will lead to the intersection of the characteristic curves at moderate mutual information level. This intersection indicates that the BER performance cannot be further improved with more iterations. At $E_b/N_0 = 1.5$ dB, the tunnel is larger, 3 outer iterations are sufficient to converge towards higher mutual information leading to lower BER. However, when performing 2 inner iterations inside the turbo decoder, the convergence point can be attained by performing 4 outer iterations. Similarly, LC-K-Best decoder in Figure 3.15b shows an equivalent performance but slightly higher E_b/N_0 is required. The convergence speed of LC-K-Best decoder is a bit lower than STS-SD, which requires more iterations to get the same performance. The reason is mainly due to the unreliability of the LLRs caused by the small list size ($\mathcal{L} = 16$). In the cases of MMSE-IC and MMSE-IC1 (Figure 3.15c and Figure 3.15d, respectively), the characteristics present a lower mutual information than the LC-K-Best decoder when $I_{A1} < 0.85$. Therefore, an equivalent performance can be obtained at higher E_b/N_0, or by performing more iterations.

In addition, the average decoding trajectory resulting from free-run iterative detection-decoding simulations is illustrated in Figure 3.15 at $E_b/N_0 = 1.5$ or 2 dB for 4 or 6 outer iterations between MIMO detector and turbo decoder with 2 inner iterations. The decoding trajectory closely matches the characteristics in the case of STS-SD and LC-K-Best decoder. The small mismatch is due to the limited interleaver length. In the case of MMSE-IC and MMSE-IC1 equalizers, the decoding trajectory diverges from the characteristics for high mutual information because the equalizer uses the *a posteriori* information to compute soft symbols instead of the extrinsic information.

Convergence of MIMO detection with LDPC decoding

In a similar way, we study the convergence of MIMO detection algorithms with LDPC decoder. Figure 3.16 shows EXIT chart of the overall system for different E_b/N_0 values and several detectors namely STS-SD, LC-K-Best decoder and MMSE-IC equalizer. The same conclusions can be obtained as in the case of turbo decoder. We can see that the convergence point is achieved at $E_b/N_0 = 1$ dB in which a clear tunnel is observed between MIMO detector and LDPC decoder characteristics allowing iterations to bring improvement to the system. Similarly, STS-SD shows the highest mutual information followed by LC-K-Best decoder and MMSE-IC equalizer.

Conclusion

In conclusion, the iterative process with large tunnel obviously leads to faster convergence. In the case of turbo decoder, if only one inner iteration is performed for each outer iteration, more than 4 outer iterations are required since the turbo decoder requires at least 6 to 8 iterations to converge. This leads to an increase of the computational complexity of the MIMO detector. If otherwise, more inner iterations are carried out (e.g, 2), 4 outer iterations in the global loop are sufficient to reach the convergence threshold with a considerable reduction of the MIMO detection complexity. In the case of LDPC decoder, we see that the convergence point can be attained by performing at least 15-20 iterations inside the LDPC decoder. The best trade-off scheduling of the required number of iterations is therefore I_{out} iterations in the outer loop and a total of 8 iterations inside the turbo decoder and 20 iterations inside the LDPC decoder distributed across these I_{out} iterations.

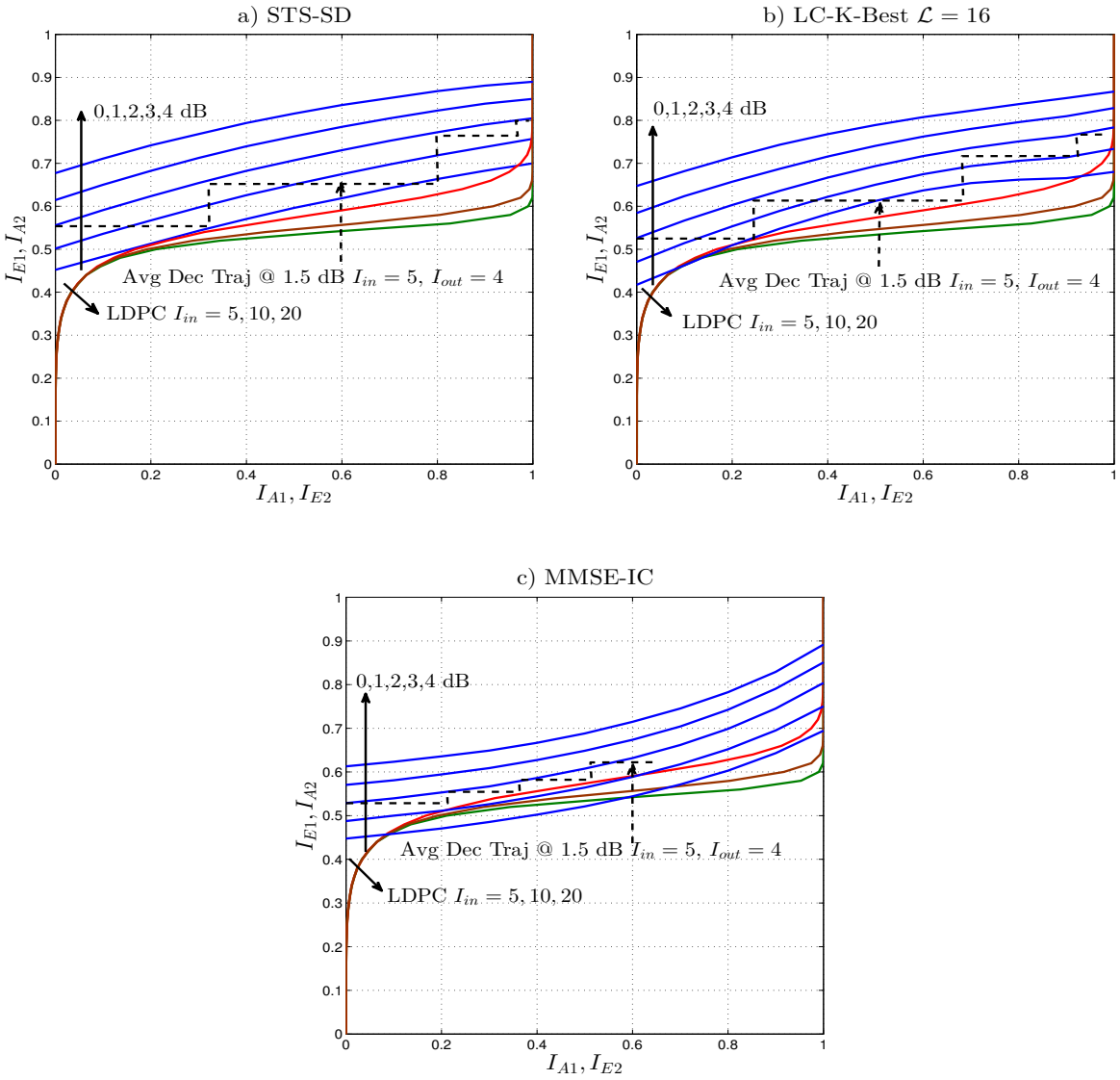


Figure 3.16: EXIT chart of LDPC decoder ($R_c = 1/2$,$N_b = 1,944$) and MIMO detectors: **(a)** STS-SD, **(b)** LC-K-Best, **(c)** MMSE-IC at different E_b/N_0 values (0, 1, 2, 3, 4 dB) in a **4 × 4** MIMO system using 16-QAM.

3.9 Performance results and discussion

In this section, we compare the performance of different MIMO detectors namely STS-SD, LC-K-Best decoder, MMSE-IC, and I-VBLAST equalizers with different channel coding (turbo, LDPC).

3.9.1 Simulation parameters

The simulations are based on a 4 × 4 SM MIMO system, QAM constellation with Gray mapping. Several modulations and coding schemes are considered to quantify the gain achieve by such an iterative receiver. The simulations are first carried out in Rayleigh fading channel to view general performance of the iterative receiver. However,

real channel models can be considered to evaluate the performance in more realistic scenarios. Therefore, 3GPP LTE(A) channel environments with low, medium, large delay spread values and Doppler frequencies are considered in the case of LTE turbo code. The low spread channel is the Extended Pedestrian A Model (EPA) which emulates the urban environment with small cell sizes ($\tau_{\text{rms}} = 43\text{ns}$). The medium spread channel is the Extended Vehicular A Model (EVA) ($\tau_{\text{rms}} = 357\text{ns}$). The Extended Typical Urban Model (ETU) is the large spread channel which has a larger excess delay ($\tau_{\text{rms}} = 991ns$) and simulates extreme urban, suburban and rural cases. Table 3.5 summarizes the characteristic parameters of these channel environments. For all cases, the channel is assumed to be perfectly known at the receiver.

Table 3.5: Characteristic parameters of the investigated channel models.

	τ_{max}	τ_{rms}	f_m	v
EPA	410 ns	43 ns	5 Hz	2 Km/h
EVA	2510 ns	357 ns	70 Hz	30 Km/h
ETU	5000 ns	991 ns	300 Hz	130Km/h

The performance is measured in terms of bit error rate (BER) with respect to SNR per information bit E_b/N_0 defined as:

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} + 10\log_{10}\frac{1}{R_c Q N_t}. \quad [dB] \tag{3.45}$$

For each E_b/N_0 value, the BER is obtained with at least 200 errors. A maximum number of $10,000$ frames is transmitted which is largely sufficient for obtaining a BER level of 10^{-5}.

For turbo code, the 1/3 rate turbo encoder specified in 3GPP LTE is used in the simulations. Puncturing is performed in the rate matching module to achieve a coding rate R_c (e.g R_c =1/2,3/4). The transmitted frame consists of either $1,024$ or $2,048$ bits. A random interleaver of a size $1,024$ or $2,048$ is therefore considered. Meanwhile, the LDPC encoder specified in IEEE 802.11n is considered. IEEE 802.11n defines three LDPC codeword lengths (648, 1296, 1944) and 4 code rates (1/2, 2/3, 3/4, 5/6) for a total of 12 possible codes. Each code is defined by a parity check matrix that is formed out of square sub-matrices of size 27, 54 or 81. Herein, the codeword length of size $1,944$ with coding rate (R_c =1/2, 3/4) is considered. The interleaver can be assumed implicitly presented in LDPC code due to its random structure. Table 3.6 summarizes the main parameters for the simulations.

In the case of K-Best decoder, LLR clipping value must be optimized in order to maximize the average mutual information at its output. The clipping level depends on the list size and the system configuration. For a small list size (e.g. 16), the clipping value was found to be relatively small between (2 and 3) to limit the error of an accurate LLR approximation at the decoder. Moreover, the list size has a significant impact on the system performance. It is obvious that if the list size increases, the performance is improved at the expense of an increase of the complexity. The list size must be then optimized for a best trade-off between performance and complexity.

Figure 3.17 shows the average mutual information at the output of LC-K-Best decoder as a function of the LLR clipping level. The results in Figure 3.17 confirm that

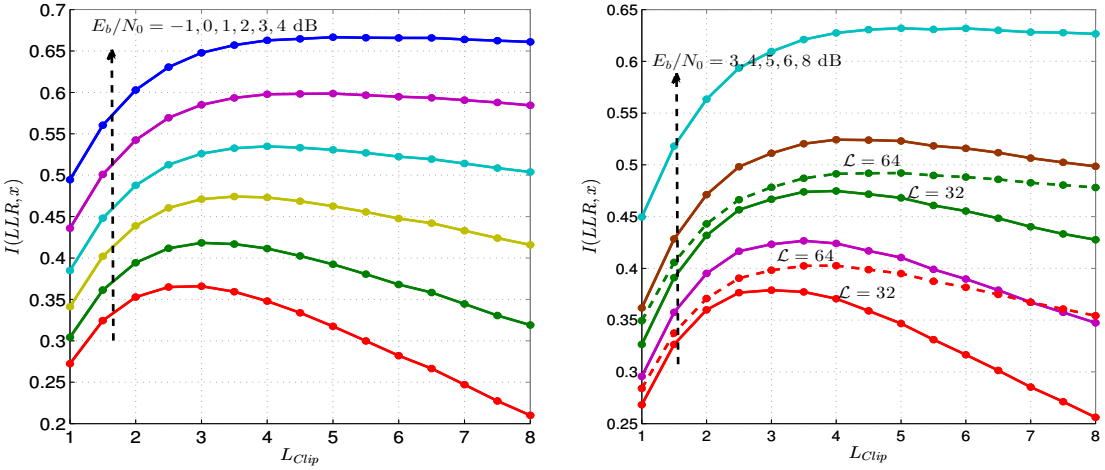
Table 3.6: Simulation parameters.

MIMO system	4×4 Spatial multiplexing
Modulation 2^Q-QAM	16-QAM, 64-QAM Gray mapping
Channel type	Flat Rayleigh fading EPA, EVA, ETU
Number of sub-carriers $N(N_c)$ Cyclic Prefix (CP) Bandwidth Carrier frequency f_c	1024 (600 usefull) samples Normal 5.2 μs (80 samples) - 4.7 μs (72 samples) 10 MHz 2.4GHz
Detector	Single tree search (STS-SD) LC-K-Best, $K = 16$ (16-QAM), $K = 32$, 64 (64-QAM) I-VBLAST, MMSE-IC, MMSE-IC1
Channel decoder	LTE turbo code $K_c = 4$ $(13, 15)_o$ $R_c = 1/2$, $3/4$ Block Length $K_b = 1,024$ or $2,048$ bits
	LDPC code (IEEE 802.11n) $R_c = 1/2$, $3/4$ Codeword Length $N_b = 1,944$ bits
Interleaver	Random, size = $1,024$, $2,048$ (turbo code case)

a clipping level between 2 and 3 give the maximum mutual information at the output in the case of 16-QAM and 64-QAM for low to medium E_b/N_0. Moreover, we see in Figure 3.17b that using a larger list size (64) in the case of 64-QAM leads to higher mutual information than using a list size of 32. A clipping level of 3 is retained in the case of LC-K-Best decoder.

3.9.2 Performance with turbo decoder

Several system configurations are considered for the iterative process to compare the performance of detection algorithms. In the first configuration, an original schedule that performs 8 inner iterations inside the turbo decoder for each outer iteration is considered. The second configuration uses a new schedule that performs a total number of 8 iterations inside the turbo decoder distributed equally across the outer iterations. Such a schedule is chosen based on the convergence behavior of the iterative processing (cf. section 3.8). For this configuration, two schemes are considered. First scheme performs 2 outer iterations, each with 4 inner iterations. The other scheme performs 4 outer iterations, each with 2 inner iterations. These schemes are considered to investigate the impact of the number of outer and inner iterations on the performance and the complexity of the system. Moreover, in the case of MMSE-IC equalizer, a previous schedule [24] that performs only one inner iteration for each outer iteration is considered. This configuration can be adopted in a low-complexity detector like the equalizer used in [24]. In the case of tree-search-based algorithms, such a configuration with 8 outer iterations requires a high computational complexity of the MIMO detector.



(a) 4 × 4-16QAM **(b)** 4 × 4-64QAM

Figure 3.17: Average mutual information $I(LLR, x)$ in function of LLR clipping level L_{clip} at different E_b/N_0 values a **4 × 4** MIMO system using LC-K-Best decoder: **(a)** 16-QAM ($\mathcal{L}$ = 16), and **(b)** 64-QAM ($\mathcal{L}$ = 32, 64).

Figure 3.18 shows the BER performance of the first configuration with $I_{in} = 8$ inner iterations and I_{out} =1,2,4, or 8 outer iterations. STS-SD is used without any simplifications which allows us to consider it as a reference close to MAP performance. At the first iteration $I_{out} = 1$ (Figure 3.18a), since no *a priori* information is available at the equalizer, a classical MMSE equalization is performed. For $I_{out} = 2, 4, 8$, an interference canceler can be carried out efficiently. Therefore, I-VBLAST, MMSE-IC equalizer and its approximation (MMSE-IC1, MMSE-IC2) are considered. From Figure 3.18a and Figure 3.18b, it can be see that the performance of the system is improved through iterations by about 1.5 dB at a BER level of 1×10^{-5} with all MIMO detection algorithms.

At the first iteration, STS-SD outperforms LC-K-Best decoder by about 0.5 dB (Figure 3.18a). However, this gap is reduced to 0.2 dB at a BER level of 1×10^{-4} with 4 outer iterations (Figure 3.18b). In addition, Figure 3.18a shows that I-VBLAST outperforms LC-K-Best decoder by 0.2 dB and 0.1 dB at a BER level of 1×10^{-5} for $I_{out} = 1$ and $I_{out} = 2$ respectively. Moreover, LC-K-Best decoder slightly outperforms MMSE equalizer by about 0.1 dB at a BER level of 1×10^{-5}. However, MMSE-IC and I-VBLAST performances are close to LC-K-Best decoder with 4 outer iterations (Figure 3.18b). MMSE-IC1 equalizer shows performance degradation of 0.4 dB compared to MMSE-IC equalizer and LC-K-Best decoder. Whereas, MMSE-IC2 presents a degradation of 0.8 dB at a BER level of 1×10^{-4} compared to MMSE-IC1. Furthermore, Figure 3.18b shows that no significant improvement can be achieved after 4 outer iterations. This improvement at a BER level of 1×10^{-4} is negligible with 8 iterations in the case of STS-SD, and less than 0.2 dB with other detectors.
We note that in most system configurations, 8 inner iterations are performed within the turbo decoder and 4 outer iterations over the MIMO detection. These numbers have been chosen since it leads to a good BER performance.

Figure 3.19 illustrates BER performance of STS-SD and LC-K-Best decoder with 4

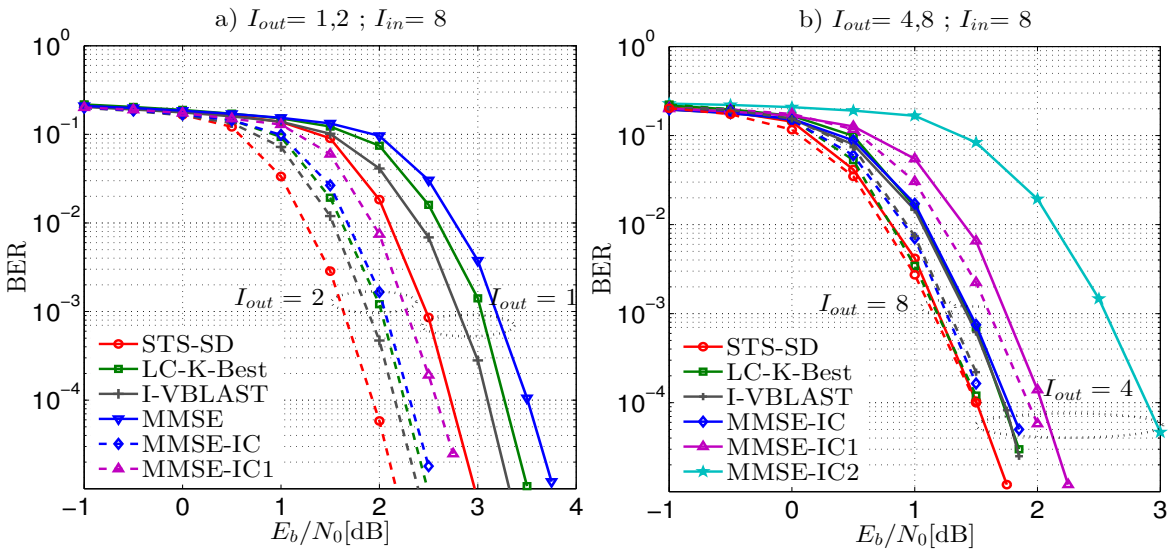


Figure 3.18: BER performance of a $\mathbf{4 \times 4}$ coded MIMO system with 16-QAM using several MIMO detectors (STS-SD, LC-K-Best, I-VBLAST, MMSE-IC and MMSE-IC1) on Rayleigh channel, **(a)** $I_{out} = 1, 2$; $I_{in} = 8$ and **(b)** $I_{out} = 4, 8$; $I_{in} = 8$. Turbo code with $R_c = 1/2$ and $K_b = 2,048$ is used.

outer iterations and 8 turbo decoder iterations distributed across these 4 outer iterations. We see that the order of inner iterations has an impact on the performance of the system. For example, performing 5 inner iterations inside the turbo decoder in the first outer iteration then 1 iteration in the remaining outer iterations shows a degradation about $0.2 \sim 0.25$ dB compared to the case when 5 inner iterations are performed in the last outer iteration. This is explained by the fact that through iterative process, the turbo decoder gets more reliable information at its input which allows faster convergence. However, this scheduling is not the optimal since the turbo decoder does not benefit from the iterations until the end. By varying the order of inner iterations, we find that a good solution is to perform 2 inner iterations inside the turbo decoder for each outer iteration. With this scheme, a degradation of 0.1 dB at 2×10^{-5} is observed with detection algorithms compared to the scheme that repeats 8 inner iterations at each outer iteration (Figure 3.18b).

Figure 3.20 illustrates the performance of MIMO detectors with the second system configuration using two different schemes. Comparing Figure 3.20a and Figure 3.20b, it can be seen that the second scheme with $I_{out} = 2$ and $I_{in} = 4$ presents a degradation of about 0.5 dB compared to the first scheme with $I_{out} = 4$ and $I_{in} = 2$. Moreover, Figure 3.20a shows that the first scheme presents a degradation of less than 0.1 dB at a BER level of 2×10^{-5} with all detection algorithms compared to the scheme that repeats 8 inner iterations at each outer iteration in Figure 3.18b. By comparing the algorithms, LC-K-Best decoder shows a degradation of less than 0.2 dB compared to STS-SD at a BER level of 2×10^{-5}. However, it outperforms MMSE-IC1 equalizer by about 0.4 dB at a BER level of 2×10^{-5}. MMSE-IC and I-VBLAST show almost the same performance as LC-KBest decoder with $I_{out} = 4$ and $I_{in} = 2$. MMSE-IC2 presents a degradation of 1 dB compared to MMSE-IC1. Moreover, with $I_{out} = 2$ and $I_{in} = 4$, I-VBLAST slightly outperforms LC-K-Best decoder and MMSE-IC equalizer.

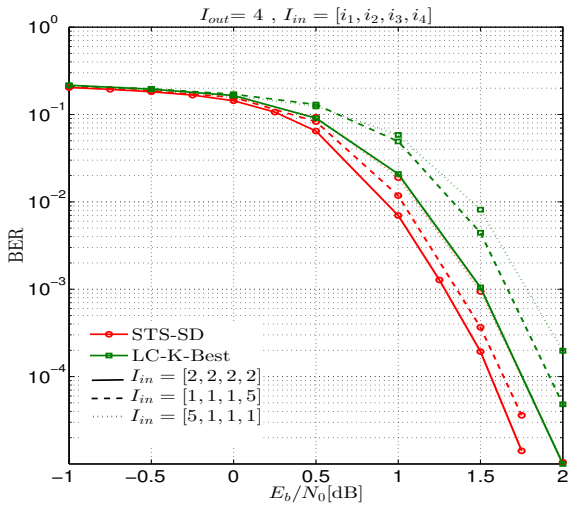


Figure 3.19: BER performance of a $\mathbf{4 \times 4}$ coded MIMO system with 16-QAM using STS-SD and LC-K-Best decoders. 8 turbo decoding iterations are distributed over 4 outer iterations. $I_{in} = [i_1, i_2, i_3, i_4]$ indicates that i_k inner iterations are performed in the k^{th} outer iteration. Turbo code with $R_c = 1/2$ and $K_b = 2,048$ is used.

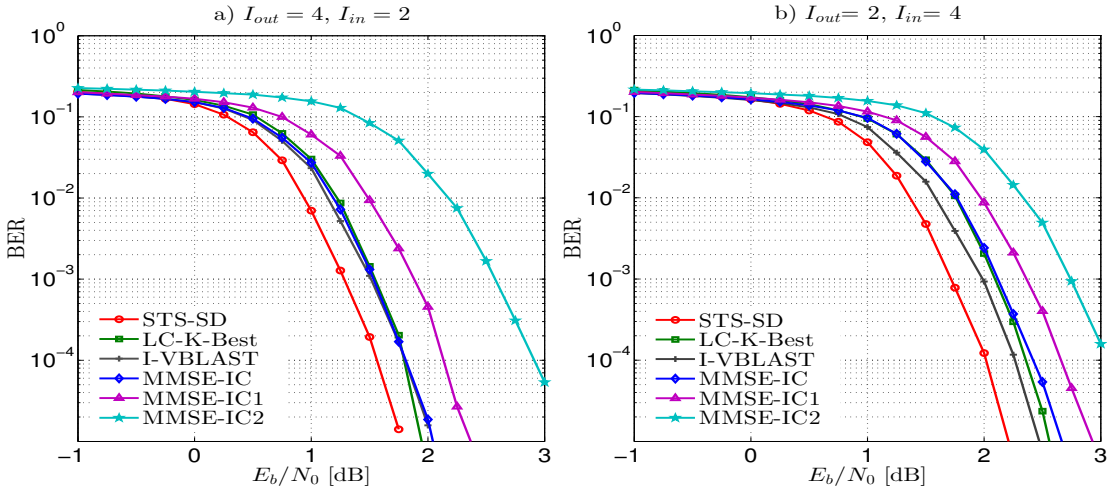


Figure 3.20: BER performance of a $\mathbf{4 \times 4}$ coded MIMO system with 16-QAM using several MIMO detectors (STS-SD, LC-K-Best, I-VBLAST, MMSE-IC, MMSE-IC1 and MMSE-IC2) on Rayleigh channel, **(a)** $I_{in} = 2$, $I_{out} = 4$ and **(b)** $I_{in} = 4$, $I_{out} = 2$. Turbo code with $R_c = 1/2$ and $K_b = 2,048$ is used.

In addition, Figure 3.21 shows the performance of MIMO detection algorithms with $I_{out} = 2, 4, 8$ and $I_{in} = 1$. We show that $I_{out} = 2$ is not sufficient for system convergence. With $I_{out} = 4$, there is a degradation of about 0.3 to 0.5 dB at the BER level of 10^{-5} compared to Figure 3.20a. However, with $I_{out} = 8$, MIMO detection algorithms present an improvement of 0.1 dB at a BER level of 2×10^{-5}. This configuration with $I_{in} = 1$ and $I_{out} = 8$ increases the complexity of the receiver especially in the case of tree-search based algorithms.

In the case of high-order modulation, higher spectral efficiency can be achieved at a

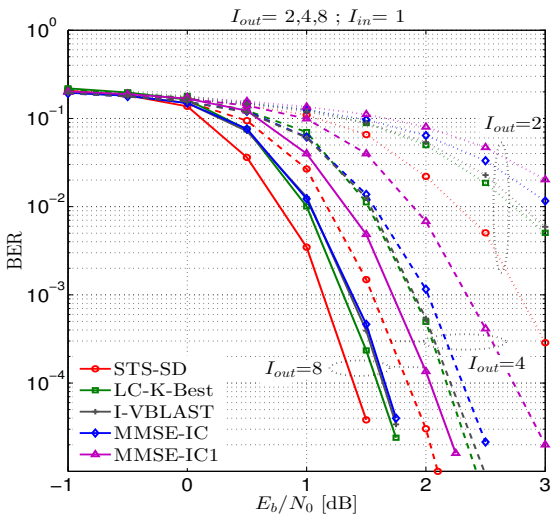


Figure 3.21: BER performance of a **4 × 4** coded MIMO system with 16-QAM using several MIMO detectors (STS-SD, LC-K-Best, I-VBLAST, MMSE-IC, MMSE-IC1 and MMSE-IC2) on Rayleigh channel, $I_{in} = 1$, $I_{out} = 2, 4, 8$. Turbo code with $R_c = 1/2$ and $K_b = 2,048$ is used.

cost of increasing of the interference which may leads to decrease the performance of the system. Therefore, an adaptive modulation and coding scheme must be used according to a given application. Figure 3.22 shows BER performance of 64-QAM with $I_{out} = 4$, $I_{in} = 2$ and R_c=3/4. We see that LC-K-Best decoder with a list size of 64 presents similar performance as STS-SD. Moreover, a performance degradation of 0.1 dB at a BER level of 1×10^{-4} is observed with a list size of 32. However, I-VLAST equalizer and MMSE-IC equalizer present a degradation of 2 dB at a BER level of 1×10^{-4} compared to LC-K-Best decoder. Therefore, LC-K-Best decoder is more robust in the case of high-order modulation and high coding rate.

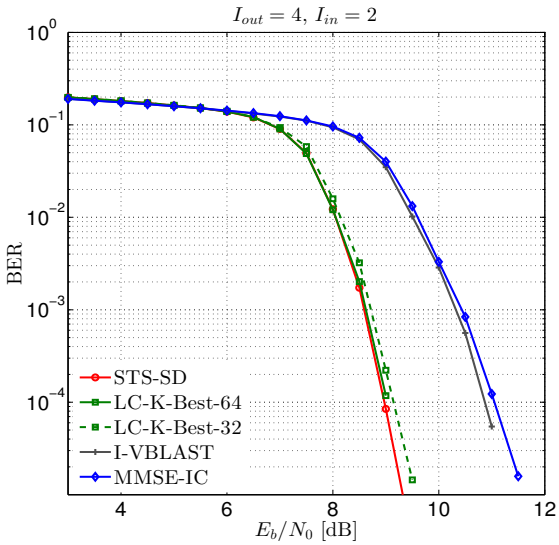


Figure 3.22: BER performance of a **4 × 4** coded MIMO system with 64-QAM using several MIMO detectors (STS-SD, LC-K-Best, I-VBLAST and MMSE-IC) on Rayleigh channel, $I_{in} = 2$, $I_{out} = 4$. Turbo code with $R_c = 3/4$ and $K_b = 2,048$ is used.

In order to summarize the performance of different detectors with different system configurations, we provide the E_b/N_0 values achieving a BER level of 2×10^{-5} in Table 3.7. The number used in the parenthesis represents the performance loss over STS-SD.

Table 3.7: E_b/N_0 values achieving a BER level of 2×10^{-5} for different detectors with $\mathbf{4 \times 4}$ 16-QAM, Turbo $R_c = 1/2$. *

	$I_{out} = 4$, $I_{in} = 8$	$I_{out} = 4$, $I_{in} = 2$	$I_{out} = 8$, $I_{in} = 1$
STS-SD	1.69 dB	1.72 dB	1.56 dB
LC-K-Best	1.87 dB (-0.18)	1.89 dB (-0.17)	1.76 dB (-0.20)
I-VBLAST	1.88 dB (-0.19)	1.97 dB (-0.25)	1.78 dB (-0.22)
MMSE-IC	1.92 dB (-0.23)	1.89 dB (-0.27)	1.76 dB (-0.24)
MMSE-IC1	2.20 dB (-0.51)	2.28 dB (-0.56)	2.20 dB (-0.64)

*The number in the parenthesis corresponds to the performance loss in dB over STS-SD.

3.9.3 Performance with LDPC decoder

In this section, we compare the performance realized by using LDPC decoder. Similarly to turbo decoder, different system configurations are considered to compare the performance of detection algorithms. Hence, we consider a first configuration that performs 20 inner iterations inside LDPC decoder for each outer iteration, and a second configuration that performs a total number of 20 iterations inside LDPC decoder distributed across outer iterations. In order to optimize the number of iterations of LDPC decoder, we fix the number of inner iterations to 20 while varying the number of outer iterations. In the next time, we fix the number of outer iterations and we change the number of inner iterations.

Figure 3.23 shows the BER performance of different detection algorithms with $I_{in} = 20$ and $I_{out} = 1, 2, 3, 4, 8$. By increasing the number of outer iterations, a performance improvement of 1.5 dB can be observed. For $I_{out} > 4$, the performance starts to saturate, the improvement is less than 0.2 dB. At the first iteration, STS-SD outperforms LC-K-Best decoder by 0.5 dB at a BER level of 10^{-4}. This gap is reduced with 4 iterations to less than 0.1 dB. Moreover, LC-K-Best decoder outperforms MMSE-IC by 0.5 dB at the first iterations and 0.25 dB with 4 iterations.

Figure 3.24 shows BER performance of detection algorithms with LDPC decoder with $I_{out} = 4$ and variable I_{in} iterations. We can see that the performance heavily depends on the number of iterations of LDPC decoder. Hence, using a large number of inner iterations does not seem to be efficient. We notice that no significant improvement can be observed if we repeat 10 or even 20 iterations in each outer iteration. This improvement is less than 0.1 dB at a BER level of 10^{-5} compared to performing a total of 20 iterations with 4 outer iterations. We notice also that increasing the number of inner iterations with each outer iterations $I_{in} = [3, 4, 6, 7]$ shows slightly better performance than performing an equal number of iterations, $I_{in} = 5$, at each outer iteration. Comparing the algorithms together, we show that STS-SD outperforms LC-K-Best by about 0.1 dB at a BER level of 2×10^{-5}. However, MMSE-IC presents a degradation of 0.25 dB compared to LC-K-Best decoder.

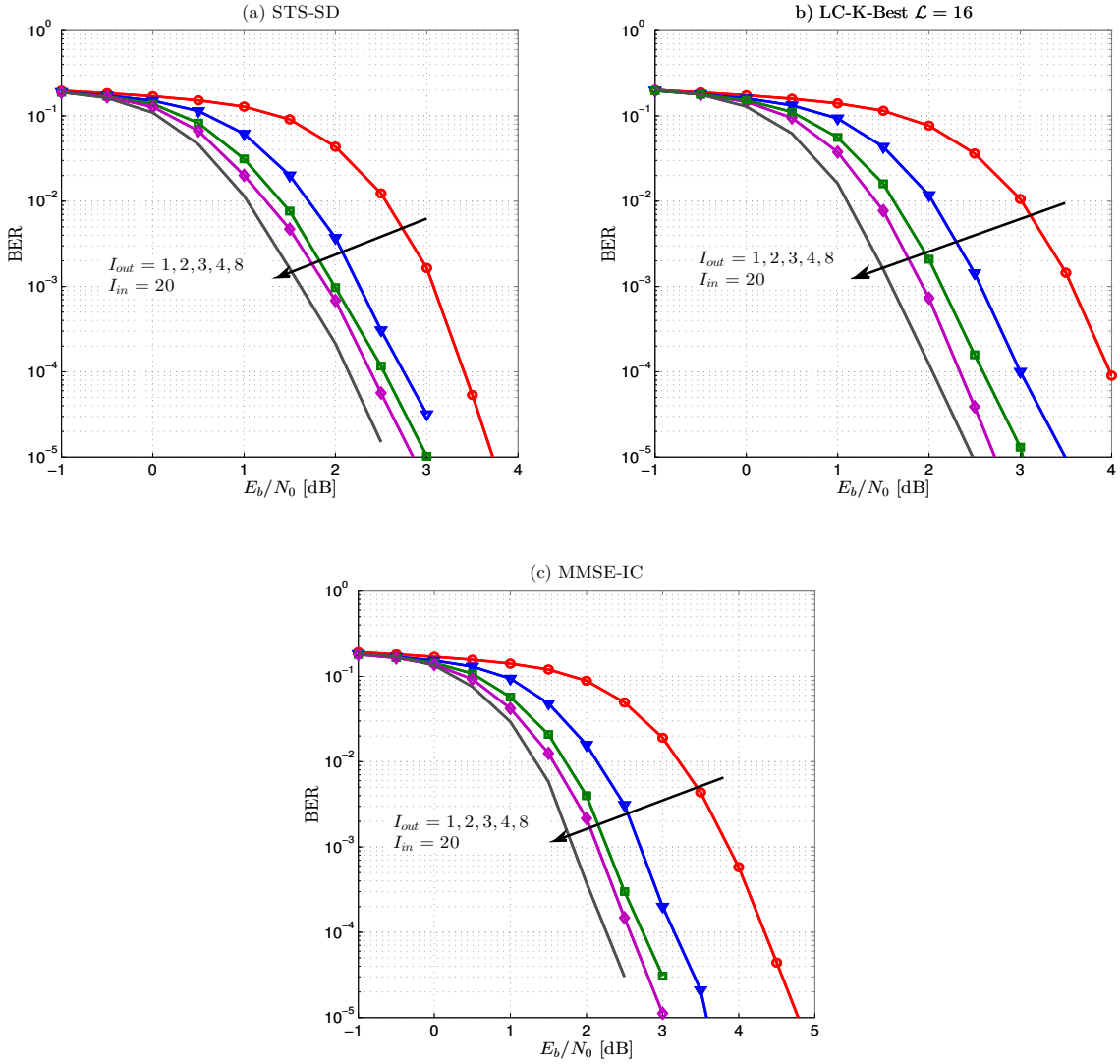


Figure 3.23: BER performance of a $\mathbf{4 \times 4}$ coded MIMO system with 16-QAM using several MIMO detectors on Rayleigh channel: (a) STS-SD, (b) LC-K-Best, (c) MMSE-IC, $I_{in} = 20$, $I_{out} = 1, 2, 3, 4, 8$. LDPC decoder with $R_c = 1/2$ and $N_b = 1,944$ is used.

Figure 3.25 shows BER performance of 64-QAM with $I_{out} = 4$, $I_{in} = 20, [3, 4, 6, 7]$ and R_c=3/4. The curves in dotted line and solid line correspond respectively to the case of $I_{in} = 20$ and $I_{in} = [3, 4, 6, 7]$. We notice that with high-order modulation and high coding rate, similar performance is achieved which confirms the inefficiency of using a large number of iterations as previously discussed. LC-K-Best decoder with a list size of 32 presents similar performance as STS-SD at high E_b/N_0. A little degradation is shown at medium E_b/N_0. However, I-VLAST equalizer and MMSE-IC equalizer present a degradation of 2.5 dB at a BER level of 1×10^{-5} compared to LC-K-Best decoder.

Table 3.8 summarizes the performance of different detectors with different system configurations in the case of LDPC decoder. The E_b/N_0 values achieving a BER level of 2×10^{-5} are considered.

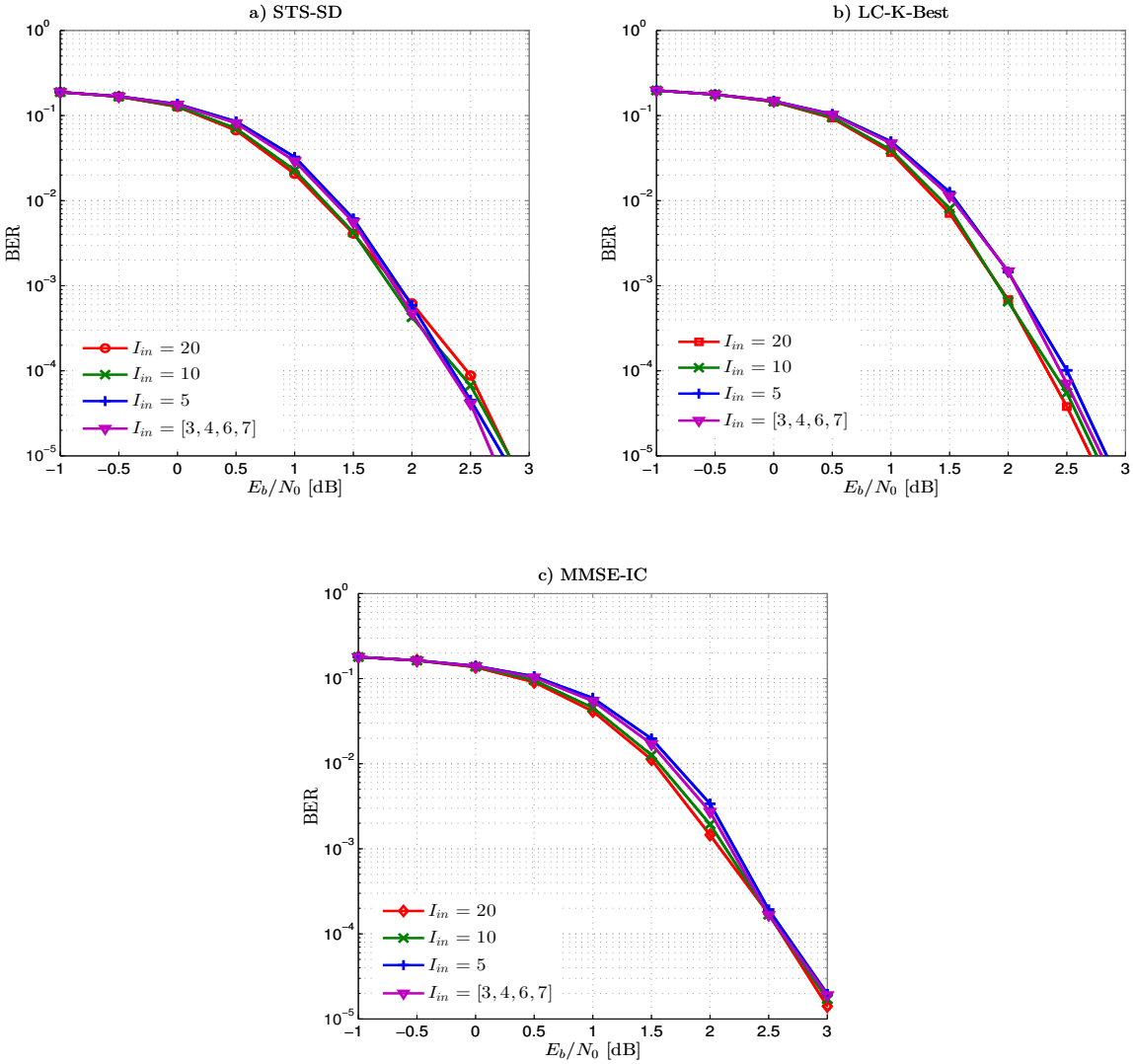


Figure 3.24: BER performance of a $\mathbf{4 \times 4}$ coded MIMO system with 16-QAM using several MIMO detectors on Rayleigh channel: (a) STS-SD, (b) LC-K-Best, (c) MMSE-IC, $I_{out} = 4$, $I_{in} = [3, 4, 6, 7], 5, 10, 20$. LDPC decoder with $R_c = 1/2$ and $N_b = 1,944$ is used.

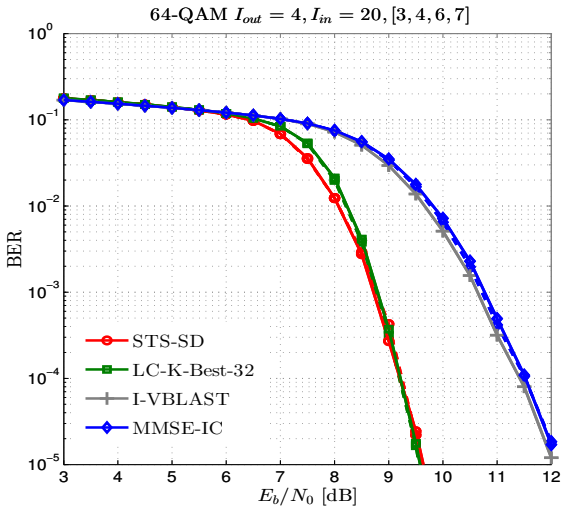


Figure 3.25: BER performance of a **4 × 4** coded MIMO system with 64-QAM using several MIMO detectors (STS-SD, LC-K-Best, I-VBLAST and MMSE-IC) on Rayleigh channel, $I_{in} = 20, [3, 4, 6, 7]$, $I_{out} = 4$. LDPC decoder with $R_c = 3/4$ and $N_b = 1,944$ is used.

Table 3.8: E_b/N_0 values achieving a BER level of 2×10^{-5} for different detectors with 4×4 16-QAM, LDPC $R_c = 1/2$.*

	$I_{out} = 4$, $I_{in} = 20$	$I_{out} = 4$, $[3, 4, 6, 7]$
STS-SD	2.70 dB	2.60 dB
LC-K-Best	2.65 dB (+0.05)	2.70 dB (-0.1)
MMSE-IC	2.90 dB (-0.2)	2.98 dB (-0.38)

*The number in the parenthesis corresponds to the performance loss in dB over STS-SD.

3.9.4 Performance with LTE channel models

In this part, we evaluate the performance of the iterative receiver in more realistic channel environments. A 4×4 SM system with 16-QAM and 64-QAM with $I_{\text{out}} = 4$ and $I_{\text{in}} = 2$ in the case of turbo decoder and $I_{\text{in}} = [3, 4, 6, 7]$ in the case of LDPC decoder is considered.

Figures 3.26, 3.27 and 3.28 show BER performance of the detectors with the channel decoders on EPA, EVA and ETU channels, receptively. Similar behaviors can be observed with LTE turbo decoder and with LDPC decoder.

In EPA channel (Figure 3.26), we see that LC-K-Best decoder achieves similar performance as STS-SD in the case of 64-AQM and presents a degradation less than 0.2 dB in the case of 16-QAM. Meanwhile, MMSE-IC presents significant performance loss of more than 6 dB in the case of 64-QAM and $R_c = 3/4$. With 16-QAM and $R_c = 1/2$, the degradation of MMSE-IC compared to LC-K-Best decoder is about 1 dB at a BER level of 1×10^{-4}.

In EVA channel (Figure 3.27), the performance loss of MMSE-IC compared to LC-K-Best decoder is reduced to approximately 5 dB with 64-QAM and 0.5 dB with 16-QAM. LC-K-Best decoder presents a degradation of about 0.1∼0.3 dB compared to STS-SD in the case of 16-QAM and 64-QAM.

Similarly in ETU channel (Figure 3.28), MMSE-IC presents a performance degradation compared to LC-K-Best decoder. This degradation is less than 4 dB in the case of 64-QAM and less than 0.5 dB in the case of 16-QAM. We notice also that LC-K-Best decoder is comparable to STS-SD in the case of 64-QAM, and has a degradation of 0.2 dB in the case of 16-QAM.

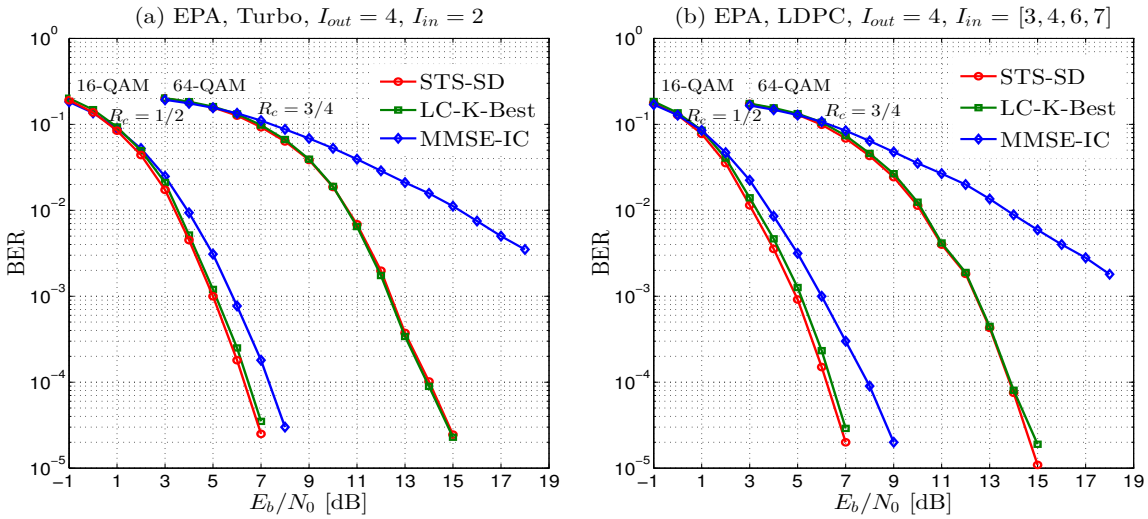


Figure 3.26: BER performance of a $\mathbf{4 \times 4}$ spatial multiplexing with 16-QAM and 64-QAM using several MIMO detectors, (STS-SD, LC-K-Best, and MMSE-IC) on **EPA** channel model, **(a)** LTE turbo decoder, $K_b = 1,024$, $I_{\text{out}} = 4$ $I_{\text{in}} = 2$, and **(b)** LDPC decoder $N_b = 1,944$, $I_{\text{out}} = 4$ $I_{\text{in}} = [3, 4, 6, 7]$.

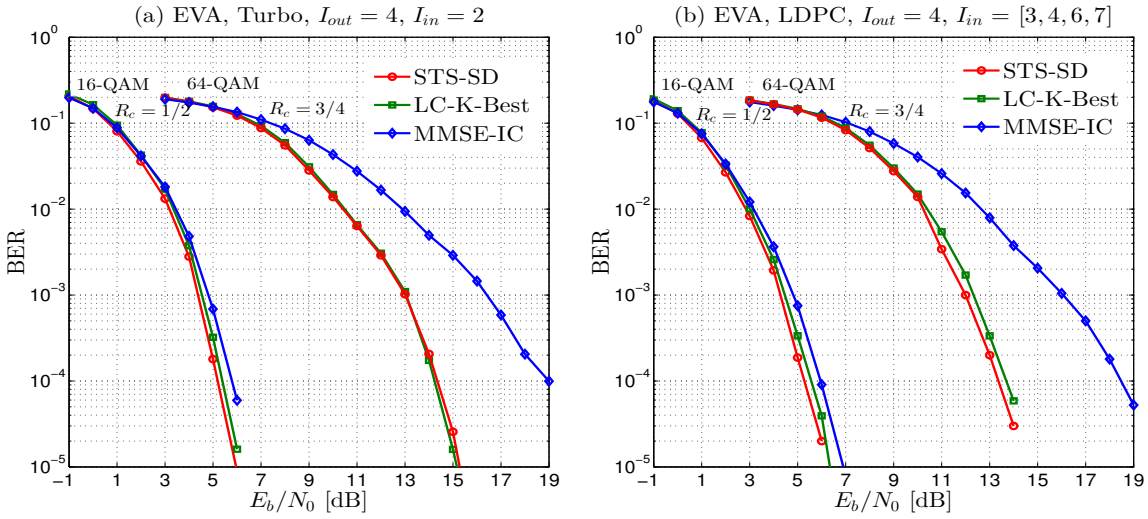


Figure 3.27: BER performance of a $\mathbf{4 \times 4}$ spatial multiplexing system with 16-QAM and 64-QAM using several MIMO detectors (STS-SD, LC-K-Best, and MMSE-IC) on **EVA** channel model, **(a)** LTE turbo decoder, $K_b = 1,024$, $I_{\text{out}} = 4$ $I_{\text{in}} = 2$, and **(b)** LDPC decoder $N_b = 1,944$, $I_{\text{out}} = 4$ $I_{\text{in}} = [3, 4, 6, 7]$.

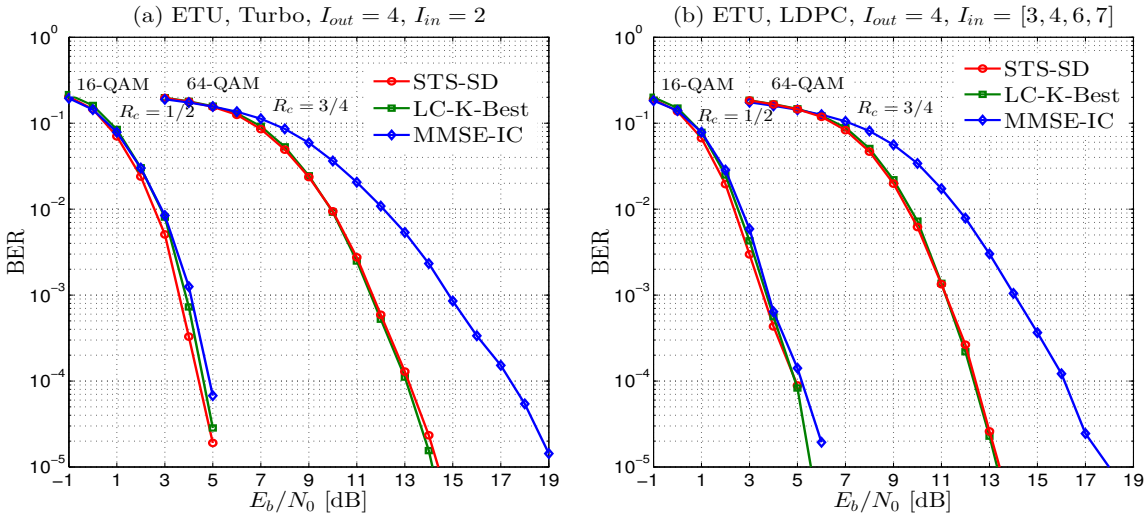


Figure 3.28: BER performance of a **4 × 4** spatial multiplexing system with 16-QAM and 64-QAM using several MIMO detectors (STS-SD, LC-K-Best, and MMSE-IC) on **ETU** channel model, **(a)** LTE turbo decoder, $K_b = 1,024$, $I_{\text{out}} = 4$ $I_{\text{in}} = 2$, and **(b)** LDPC decoder $N_b = 1,944$, $I_{\text{out}} = 4$ $I_{\text{in}} = [3, 4, 6, 7]$.

Comparing the performance of the iterative receiver in different channels, it can be seen that the iterative receiver presents best performance in ETU channel compared to EPA and EVA channels. This is due to the high diversity of ETU channel. At a BER level of 1×10^{-4}, the performance gain in ETU channel in the case of LTE turbo decoder is about 0.8 dB, 1.3 dB with 16-QAM and 64-QAM respectively compared to EVA channel. In the case of LDPC decoder, this gain is 0.4 dB and 1 dB with 16-QAM and 64-QAM receptively. However, in EPA channel, the performance gain in ETU channel in the case of turbo decoder or LDPC decoder is more than 1 dB with 16-QAM and 64-QAM.

Table 3.9 summarizes the E_b/N_0 values achieving a BER level of 1×10^{-4} of different detectors combined with different channel decoders, modulation orders in different channel models. The values given in the parentheses in the table represent the performance loss compared to STS-SD. As indicated in the table, the iterative receiver with turbo decoder and LDPC decoder has a comparable performance with a coding rate $R_c = 1/2$ (16-QAM). However, with $R_c = 3/4$ (64-QAM), the receiver with LDPC decoder presents slightly a better performance, especially in ETU channel (0.6 dB).

From these results, we show that iterative receiver substantially improves the performance of coded MIMO systems with both turbo decoder and LDPC decoder in Rayleigh channel (Figure 3.18, 3.23) and in more realistic channels (Figure 3.26, 3.27, 3.28). Moreover, after a certain number of iterations, the performance of the system becomes saturated and does not show significant improvement anymore. Additionally, Figure 3.19, Figure 3.20 and Figure 3.24 show that the scheduling order and the number of iterations affect the system performance. We show that performing a large number of inner iterations does not bring significant improvement. The figures suggest that BER performance of the iterative receiver with turbo decoder is almost comparable to that of LDPC decoder. It is therefore meaningful to evaluate the computational com-

plexity of the iterative receiver with both decoding techniques as it will be discussed in the next chapter.

Table 3.9: $\boldsymbol{E_b/N_0}$ values achieving a BER level of $\mathbf{1 \times 10^{-4}}$ in LTE channel models for different detectors and channel decoders (turbo, LDPC) in $\mathbf{4 \times 4}$ spatial multiplexing system with 16-QAM $\boldsymbol{R_c = 1/2}$, and 64-QAM $\boldsymbol{R_c = 3/4}$.*

		Turbo code		LDPC code	
		16-QAM	64-QAM	16-QAM	64-QAM
EPA	STS-SD	6.3 dB	14.0 dB	6.2 dB	13.8 dB
	LC-K-Best	6.5 dB (−0.2)	13.9 dB (+0.1)	6.4 dB (−0.2)	13.9 dB (−0.1)
	MMSE-IC	7.4 dB (−1.1)	> 20 dB (> −6)	8.0 dB (−1.8)	> 20 dB (> −6)
EVA	STS-SD	5.2 dB	14.3 dB	5.3 dB	13.4 dB
	LC-K-Best	5.4 dB (−0.2)	14.4 dB (−0.1)	5.6 dB (−0.3)	13.7 dB (−0.3)
	MMSE-IC	5.8 dB (−0.6)	19.0 dB (−4.7)	6.0 dB (−0.7)	18.5 dB (−5.1)
ETU	STS-SD	4.4 dB	13.0 dB	4.9 dB	12.4 dB
	LC-K-Best	4.6 dB (−0.2)	13.0 dB (0.0)	4.9 dB (0.0)	12.4dB (0.0)
	MMSE-IC	4.9 dB (−0.5)	17.5 dB (−3.5)	5.1 dB (−0.2)	16 dB (−3.6)

*The number in the parenthesis corresponds to the performance loss in dB compared to STS-SD.

3.10 Conclusion

In this chapter, iterative receiver combining multiple-input multiple-output (MIMO) detection with channel decoding has been considered in order to achieve near-capacity performance and reliable high data rate transmission, for future wireless communication systems.

We first presented an overview of the soft-input soft-output MIMO detection algorithms. Thus, a low-complexity K-Best decoder (LC-K-Best) is proposed to avoid the full expansion and to simplify the enumeration through the use of two LUTs. Moreover, we analyze the convergence of combining these detection algorithms with the channel decoder using the extrinsic information transfer (EXIT) chart. Based on this analysis, we retrieved the required number of inner/ outer iterations required for the convergence of the iterative receiver. Simulation results show that the proposed K-Best decoder achieves a good performance-complexity tradeoff. We also show that the new scheduling gives similar performance as the original scheduling while saving a large amount of complexity and latency.

However, the iterative processing increases the computational complexity at the receiver. In the next chapter, the computational complexity of MIMO detection algorithms combined with channel decoding will be investigated.

Chapter 4

Computational Complexity and Fixed-Point Arithmetic

Contents

We have seen in the previous chapter that iterative receiver improves the performance of MIMO systems. However, the main challenge of such a receiver is its computational complexity and its implementation aspects which will be the topic of this chapter.

In practical systems, the computational complexity impacts the latency, the throughput, and the power consumption of the device. Therefore, the receiver algorithms must be a compromise between performance and cost. A careful performance versus complexity comparison is required to choose an optimized algorithm for a given application. In this chapter, we carry out a detailed study of the computational complexity of MIMO detection algorithms and channel decoders (turbo, LDPC). To this end, the overall complexity of the iterative receiver for different system configurations are compared.

In addition, floating-point arithmetic involves extra cost in hardware implementation. However, fixed-point arithmetic can significantly improve the execution speed and reduce the material cost by reducing the range and the accuracy of algorithm variables. However, finite word length can degrade the performance of the system due to the overflow and the quantization noise. Therefore, we have to find an efficient fixed-point representation of system parameters for the iterative receiver with negligible performance degradation.

4.1 Complexity assumptions

The concept of complexity has several significations in the system design.
At the algorithm level, we have the computational complexity expressed in terms of the numbers and types of floating-point operations (addition/subtraction, multiplication, division...). Generally $\mathcal{O}$ notation is widely used to give an asymptotic approximation of the computational complexity.

At the architectural level, more metrics are used to give an estimation of the complexity. The metrics correspond to the hardware complexity and depend on the type of implementation which can be either based on software or hardware processor. The hardware complexity represents the area or resources required for the implementation. In FPGA implementation for example, the area is expressed by the number of LUTs, registers, memory blocs and arithmetic blocs. While in a DSP based architecture, the area corresponds to the number of processing units.

Execution time or running time is another indication of the computational complexity. The execution time is also referred to as time latency in the literature.
The computational and the material complexity affect the cost and the power consumption of any application and must be therefore reduced for an efficient architecture in terms of area, power consumption and latency.

Through our analysis, we evaluate the computational complexity of the iterative receiver in terms of basic real-valued operations such as addition, subtraction, multiplication, division, square root extraction, maximization, as well as look-up-table check (which are denoted by ADD, SUB, MUL, DIV, SQRT, Max, and LUT, respectively). Several assumptions and general rules are used in the computation. The complex operations are transformed into an equivalent number of real operations using Table 4.1. The subtraction is assumed to have the same level of complexity as the addition (use of two's complement).

Table 4.1: Equivalent real number of complex operations.

Complex Operations	Real ADD	Real MUL
Complex ADD	2	0
Complex MUL	2	4
Matrix MUL A(p,n) B(n,m)	4pnm-2pm	4pnm

4.2 Iterative receiver complexity

For an iterative receiver, the algorithm complexity is related to the channel decoder, the MIMO detector and the number of iterations. The overall complexity of an iterative receiver can be expressed by:

$$C_{\text{total}} = I_{\text{in}} I_{\text{out}} C_{\text{dec}} N_{\text{bit}} + N_{\text{symb}} \left\{ C_{\text{det1}} + (I_{\text{out}} - 1) C_{\text{deti}} \right\}, \tag{4.1}$$

where C_{det1} denotes the complexity of the first iteration of MIMO detection algorithm per symbol vector without taking into consideration the *a priori* information. While

C_{deti} denotes the complexity per iteration and per symbol vector taking into consideration the *a priori* information. C_{dec} denotes the complexity of the channel decoder (turbo or LDPC) per iteration and per information bit. N_{bit} is the number of information bits at the input of the encoder, while N_{symb} is the number of symbol vectors. N_{symb} and N_{bit} are linked by the following relation:

$$N_{\text{symb}} = \frac{N_{\text{bit}}}{QR_cN_t} = \alpha N_{\text{bit}}, \quad \text{with} \quad \alpha = \frac{1}{QR_cN_t}, \tag{4.2}$$

where Q is the number of bits in the constellation symbol, R_c is the coding rate and N_t is the number of transmit antennas.

4.3 Turbo decoder complexity

The turbo decoder complexity has been widely investigated in the literature. This complexity depends on SISO decoder algorithms and number of iterations. Herein, max-log-MAP algorithm with a correction factor is used [5]. In this section, the computational complexity of turbo decoder per information bit is evaluated. Its total complexity can be hence obtained by multiplying it by the block length K_b and by the number of iterations $I_{\text{in}}I_{\text{out}}$.

Complexity of max-log-MAP decoder

The complexity of max-log-MAP decoder corresponds to three principal computations: branch metrics, recursive state metrics, and LLRs of the bits. The required number of operations can be computed as follow.

Branch metrics: At each instant k, the decoder receives systematic information, parity information and *a priori* information about systematic bit. For a binary code rate $R = 1/n$, the branch metric corresponds to the transition probability of 2^n possible bit combinations. For example, with R=1/2, there are 4 possible combinations of systematic and parity bits (00, 01, 10, 11). Therefore, the required number of operations is:

- 3 Memory accesses for systematic, parity and *a priori* information.
- 3 ADD operations to compute possible values of branch metric $\gamma_k(s_{k-1}, s_k)$.

Recursive state metrics: The computation of state metric α or β requires 2 additions and one maximum operations for each state. For 2^m trellis states, where m is the memory length of the component encoder, the required number of operations per information bit is as follow:

- 2.2^m ADD operations to compute $\alpha_{k-1}(s_{k-1}) + \gamma_k(s_{k-1}, s_k)$ or $\beta_{k+1}(s_{k+1}) + \gamma_k(s_k, s_{k+1})$ that correspond to 2.2^m trellis transitions (2^m trellis states).
- 2^m Max operations with 2 input ports, Max(2-input), to compute α_k or β_k.
- 2^m LUT to find the correction term, $\log\left(1 + e^{-|a-b|}\right)$, in max* operation.
- 2^m Store operations to write either α_k or β_k metrics in the memory.
- 2^m SUB operations to normalize the state metrics by subtracting the maximum metric from all other metrics.

Log likelihood ratio: For iterative decoding, extrinsic information of all bits (systematic and parity) are required. The computation of LLRs can be done along with either forward or backward metrics, therefore two more additions are needed at each state to compute the second addition in equation (1.26). Then maximum operations with 8 input ports is required to find the maximum of all states. The complexity is therefore:

- 2.2^m ADD operations to compute the second additions for the 2.2^m trellis transitions.
- 2^m Memory accesses for either forward or backward metrics.
- $2n$ Max operations with 2^m input ports, Max (2^m-input), to compute the maximum when the bit is 0 and 1. We note that maximum operations with N inputs is equivalent to (N-1) maximum operations with 2 inputs. Therefore the required number of Max operations is $2n(2^m - 1)$ Max (2-input).
- $2n(2^m - 1)$ LUT to find the correction term, $\log\left(1 + e^{-|a-b|}\right)$, in max* operations.
- n SUB operations to get the *a posteriri* LLR.
- 1 SUB operation to get the extrinsic LLR for information bit.
- n SUB operations to get the extrinsic LLR for coded bits.
- n Store operations to write the extrinsic LLR in the memory.

Table 4.2 summarizes the total number of operations per info bit of the max-log-MAP algorithm.

Table 4.2: Complexity of max-log-MAP algorithm.

	ADD/SUB	Max (2-input)	LUT
Gamma	$2^{n-1}(n+1) - 2^n + 1$	0	0
Alpha	4.2^m	2^m	2^m
Beta	4.2^m	2^m	2^m
LLR	$2.2^m + 2.n.2^m + 1$	$2.n.(2^m - 1)$	$2.n.(2^m - 1)$
max-log-MAP	$2^{n-1}(n+1) - 2^n + 10.2^m + 2.n.2^m + 2$	$2.2^m + 2.n.(2^m - 1)$	$2.2^m + 2.n.(2^m - 1)$

For LTE turbo decoder with $2^m = 8$ states and $n = 2$ output bits, the total number of operations per info bit and per iteration is presented in Table 4.3.

Table 4.3: Complexity of LTE turbo decoder per information bit per iteration.

	ADD	Max (2-input)	LUT
Turbo decoder $2^m = 8, n = 2$	232	88	88

4.4 LDPC decoder complexity

Sum-product algorithm (SPA) described in section (1.3.2.2) can be reformulated to reduce the computational complexity of the decoder. The equation (1.35) can be written as:

$$L_{c_{ji}} = \left(\prod_{i\prime \in V_j \backslash i} \left(\text{sign}(\text{L}_{\text{v}_{i\prime j}}) \right) \right) \varphi \left(\sum_{i\prime \in V_j \backslash i} \varphi \left(\left| L_{v_{i\prime j}} \right| \right) \right), \tag{4.3}$$

where the function $\varphi(x)$ is defined by:

$$\varphi(x) = -\log\left(\frac{e^x - 1}{e^x + 1}\right) = -\log\left(\tanh\left(x/2\right)\right). \tag{4.4}$$

The product of the sign can be calculated using modulo 2 addition, while the function $\varphi(x)$ can be easily implemented using a look-up table (LUT). The complexity of SPA is therefore simplified into additions and look-up tables.
Furthermore, LDPC decoder complexity depends on the scheduling used to exchange the messages between CNs and VNs. There are two distinct schedules of belief propagation algorithm: flooding schedule and layered schedule. In the flooding schedule, the messages are passed back and forth along all the edges. This schedule increases the complexity especially for longer block length. A layered schedule is hence proposed where only a small number of check nodes and variable nodes are updated per sub-iteration [177]. The messages generated in a sub-iteration are immediately used in subsequent sub-iterations of current iteration. This leads to faster convergence of LDPC decoding and reduction of the required memory size.
Let d_{vi} and d_{cj} denote the degree of connectivity of variable node i and check node j, respectively. The computational complexity of layered LDPC decoder can be expressed as a function of the degree of connectivity as summarized in Table 4.4. d_c and d_v denote the average row weight and the average column weight of LDPC code, respectively.

Table 4.4: Complexity of LDPC decoder.

	ADD/SUB	LUT
CN update	$\sum_{j=1}^{M_b}(2d_{cj}-1) + \sum_{i=1}^{N_b} d_{vi} = M_b(2d_c - 1) + N_b d_v$	$\sum_{j=1}^{M_b}(2d_{cj}) = 2M_b d_c$
VN update	$\sum_{i=1}^{N_b} d_{vi} = N_b d_v$	-

4.5 MIMO detection complexity

The computational complexity of MIMO detection depends on the detection algorithm. In the case of tree-search-based algorithms, the commonly approach used to measure the complexity is to count the number of visited nodes in the tree-search process. However, in the case of the interference cancellation-based equalizers, the complexity is evaluated in terms of real or complex operations required to compute filter coefficients. For a fair comparison, the complexity is estimated based on basic operations (ADD, SUB, MUL, DIV, SQRT, Max, and LUT) in this section.

4.5.1 MAP algorithm complexity

MAP algorithm performs an exhaustive search over $2^{N_t Q}$ possible symbol vectors to minimize the metric $\|\mathbf{y} - \mathbf{Hs}\|^2$. The computational complexity of MAP algorithm per each detected symbol vector is summarized in Table 4.5.

4.5.2 MMSE-IC equalizer complexity

The complexity of MMSE-IC equalizers is dominated by the computation of filter coefficients and matrix inversion. Several methods for matrix inversion have been widely

Table 4.5: Complexity of MAP algorithm per detected symbol vector.

	MUL	ADD	Min (2-input)
$\mathbf{Hs}$	$2^{N_tQ}4N_tN_r$	$2^{N_tQ}\left(4N_tN_r - 2N_r\right)$	-
$\|\mathbf{y}-\mathbf{Hs}\|$	-	$2^{N_tQ}2N_r$	-
$\|\mathbf{y}-\mathbf{Hs}\|^2$	$2^{N_tQ}\left(2N_r\right)$	$2^{N_tQ}\left(2N_r - 1\right)$	-
$\min\{.\}$	-	N_tQ	$2N_tQ\left(2^{N_tQ}-1\right)$
MAP	$2^{N_tQ}\left(4N_tN_r + 2N_r\right)$	$2^{N_tQ}\left(4N_tN_r + 2N_r - 1\right) + N_tQ$	$2N_tQ\left(2^{N_tQ}-1\right)$

studied in the literature namely Cholesky decomposition and QR decomposition. The complexity of matrix inversion of dimension $N \times N$ based on Cholesky decomposition is approximately $\mathcal{O}(N^3)$. Herein, QR decomposition based on Gram-Schmidt method is used to help the computation of the matrix inversion (Table 4.6). However, more efficient method for QR decomposition may be considered to optimize the cost of computational complexity in hardware implementation, like Givens rotations (GR) that can be effectively done by coordinate rotation digital computer (CORDIC) scheme under triangular systolic array.

Table 4.6: Complexity of matrix inversion based on QR decomposition.

Expression	MUL	ADD	DIV	SQRT
QRD $\mathbf{A}(N_t, N_r)$	$4N_t^2N_r - 2N_tN_r$	$4N_t^2N_r - 2N_tN_r - N_t^2$	$2N_tN_r$	N_t
$\mathbf{R}^{-1}$	$\frac{2}{3}N_r^3 - \frac{5}{3}N_r + 1$	$\frac{2}{3}N_r^3 - 2N_r^2 + \frac{4}{3}N_r$	N_r	-
$\mathbf{A}^{-1} = \mathbf{R}^{-1}\mathbf{Q}^H$	$2N_r^2N_t + 2N_tN_r$	$2N_r^2N_t$	-	-

The different steps for computing the coefficient of the filters and the equalized symbol vector with their corespondent complexities are reported in Table 4.7, 4.8, 4.9, 4.10 for MMSE, MMSE-IC, MMSE-IC1 and MMSE-IC2 respectively [178].

Table 4.7: Complexity of MMSE algorithm for 1st iteration per symbol vector.

Expression	MUL	ADD	DIV	SQRT
$\mathbf{H}^H\mathbf{H} + \sigma_n^2\mathbf{I}_{N_t}$	$4N_t^2N_r$	$4N_t^2N_r - 2N_t^2 + N_t$	-	-
$(...)^{-1}$	matrix inversion (Table 4.6)			
$\mathbf{p}_i$	$4N_t^2N_r$	$4N_t^2N_r - 2N_tN_r$	-	-
$\tilde{s}_i$	$4N_tN_r$	$4N_tN_r - 2N_t$	-	-
β_i	$4N_tN_r$	$4N_tN_r - 2N_t$	-	-
$\sigma_{\eta_i}^2$	$2N_t$	N_t	-	-

4.5.3 Soft mapper and soft demapper complexity

Soft mapper converts the *a priori* information coming out from the channel decoder into soft estimated symbols. We have seen that when using the Gray mapping, the real and imaginary parts of each symbol are independently mapped. This mapping can be exploited to reduce the computational complexity of soft estimated symbols that can

be computed with simple additions and multiplications [26]. The complexity of soft mapper is summarized in Table 4.11 for different modulation orders.

Table 4.8: Complexity of MMSE-IC algorithm for i^{th} iteration per symbol vector.

Expression	MUL	ADD	DIV
$\mathbf{H}\mathbf{V}_i\mathbf{H}^H + \sigma_n^2\mathbf{I}_{N_r}$	$4N_r^2N_t^2 + 2N_rN_t^2$	$4N_r^2N_t^2 - 2N_r^2N_t + N_rN_t$	-
$(...)^{-1}$	matrix inversion (Table 4.6)		
$\mathbf{p}_i$	$4N_r^2N_t + N_rN_t$	$4N_r^2N_t - 2N_rN_t$	-
$\mathbf{q}_i$	$4N_rN_t^2$	$4N_rN_t^2 - 2N_t^2$	-
$\tilde{s}_i$	$4N_rN_t + 4N_t^2$	$4N_rN_t + 4N_t^2 - 3N_t$	-
β_i	$4N_rN_t$	$4N_rN_t - 2N_t$	-
$\sigma_{\eta_i}^2$	$2N_t$	N_t	-

Table 4.9: Complexity of MMSE-IC1 algorithm for i^{th} iteration per symbol vector.

Expression	MUL	ADD	DIV
$\mathbf{H}\mathbf{H}^H(\sigma_s^2 - \sigma_{\hat{s}}^2) + \sigma_n^2\mathbf{I}_{N_r}$	$4N_r^2N_t + 2N_r^2$	$4N_r^2N_t - 2N_r^2 + N_r$	-
$(...)^{-1}$	matrix inversion (Table 4.6)		
$\bar{\mathbf{p}}_i$	$4N_r^2N_t + N_rN_t$	$4N_r^2N_t - 2N_rN_t$	-
λ_i	$4N_rN_t + N_t$	$4N_rN_t - N_t$	N_t
$\mathbf{p}_i$	$2N_rN_t$	-	-
$\mathbf{q}_i$	$4N_rN_t^2$	$4N_rN_t^2 - 2N_t^2$	-
$\tilde{s}_i$	$4N_rN_t + 4N_t^2$	$4N_rN_t + 4N_t^2 - 3N_t$	-
β_i	$4N_rN_t$	$4N_rN_t - 2N_t$	-
$\sigma_{\eta_i}^2$	$2N_t$	N_t	-

Table 4.10: Complexity of MMSE-IC2 algorithm for i^{th} iteration per symbol vector.

Expression	MUL	ADD	DIV
$\mathbf{p}_i$	$4N_rN_t + N_t$	N_rN_t	N_t
$\mathbf{q}_i$	$4N_rN_t^2$	$4N_rN_t^2 - 2N_t^2$	-
$\tilde{s}_i$	$4N_rN_t + 4N_t^2$	$4N_rN_t + 4N_t^2 - 3$	-
β_i	$4N_rN_t$	$4N_rN_t - 2N_t$	-
$\sigma_{\eta_i}^2$	$2N_t$	N_t	-

Table 4.11: Complexity of soft Gray mapper per symbol vector.

Constellation	MUL	ADD	LUT(tanh)
(4|16|64)-QAM	[2|4|6] N_t	[−|2|4] N_t	QN_t

Similarly, the LLR can be efficiently computed in case of Gray mapping by simple subtraction, multiplication and comparison [26]. Table 4.12 summarizes the complexity of soft demapper with different QAM constellations.

Table 4.12: Complexity of soft Gray demapper per symbol vector.

Constellation	MUL	DIV	ADD
(4|16|64)-QAM	$[3\|7\|9]\,N_t$	N_t	$[-\|4\|6]\,N_t$

4.5.4 Sphere decoder complexity

The complexity of tree-search-based algorithms can be divided into two steps: the preprocessing step and the tree-search step. SD complexity has been widely discussed in the literature [81, 79, 80, 86, 91]. This complexity depends on the number of transmit antennas and modulation orders. The average complexity of SD has been shown to be polynomial in the number of unknowns roughly $\mathcal{O}(N_t^3)$. However, it presents an exponential complexity in the worst case conditions depending on the noise level and the choice of an initial radius. Due to the sequential nature of the tree-search and the statistical effect of the channels, it is very difficult to find an analytical expression of SD complexity. Empirical results have been then used by assuming that the complexity is a random variable with polynomial expected complexity. Several approaches have been used to evaluate SD complexity. One approach consists in measuring the number of floating-point operations [81, 79]. The second way is to measure the expected number of visited nodes during the tree-search [86] since this number is directly proportional to the number of operations.

We note also that SD complexity depends on the preprocessing steps including QR decomposition and initial value of the radius. In the case of quasi-stationary channel, the channel matrix is assumed to be constant over a long period of time. QR decomposition can be performed only once over the frame. Therefore, their associated complexity can be substantially reduced in slow fading environment. Moreover, QR decomposition is performed at the first iteration. The complexity of QR decomposition can be then considered negligible compared to the global complexity of the iterative receiver. For the initial radius choice, if a deterministic way is used such as ZF or MMSE estimates, then the complexity of the radius estimation must be taken into account.

The authors in [79, 86] give an expression of SD complexity for infinite and finite lattices in terms of the number of flops and the number visited nodes in the sphere of radius r_s and dimensions $i = 1, 2, .., m_s$. This complexity is expressed as:

$$C\left(m_s, \sigma_n^2, r_s^2\right) = \sum_{i=1}^{m_s} \Big(\text{expected number of points in } i^{\text{th}} \text{ sphere of radius } r_s\Big) . f_p\left(i\right), \tag{4.5}$$

where $f_p\left(i\right)$ represents the number of elementary operations (additions, subtractions and multiplications) per each visited node in dimension i; and it is given by $f_p\left(i\right) = 2i + 11$ for infinite lattice. In [79, 86], expressions for different finite lattices have been proposed. The expression (4.5) states that the complexity is a function of the sphere radius r_s, the SNR (σ_n^2) and the number of antennas (tree dimension m_s). Therefore for a large radius and a low SNR, more points lie inside the sphere and the complexity becomes exponential. The variable complexity of SD leads to a variable throughput, which makes it inefficient for practical implementation. For this reason, the research

tends to several variants of SD algorithm such as K-Best decoder and Fixed sphere decoder in order to reduce the complexity.

Herein, Monte Carlo simulations were used to measure the number of operations (additions and multiplications) of sphere decoder specifically STS-SD.

4.5.5 K-Best decoder complexity

K-Best decoder was proposed to overcome the variable complexity of SD by retaining a fixed number of nodes in each level using a breadth-first search. K-Best algorithm has a fixed complexity regardless of the channel environment. Its complexity depends on the number of transmit antennas, the modulation order and the number of retained candidates at each level. The computational complexity of preprocessing step (SQRD), the detection step (K-Best and LC-K-Best) for $N_t \times N_r$ 2^Q-QAM in real system model are summarized in Tables 4.13 and 4.14, respectively.

Table 4.13: Complexity of preprocessing step (SQRD).

Expression	MUL	ADD	DIV	SQRT
SQRD [**QR**]	$8N_t^2N_r + 2N_t^2 - N_t$	$8N_t^2N_r - 2N_t$	$4N_tN_r$	$2N_t$
$\tilde{\mathbf{y}}$	$4N_tN_r$	$4N_tN_r - 2N_t$		

Table 4.14: Complexity of K-Best and LC-K-Best algorithm per symbol vector.

Expression	MUL	ADD
K-Best 1st itr	$2^{Q/2}K(2N_t^2+3N_t-2)+$ $2.2^{Q/2}$	$2^{Q/2}K(2N_t^2+3N_t-2)+$ $2^{Q/2}$
K-Best i^{th} itr	$2^{Q/2}K(2N_t^2+3N_t-2)+$ $2.2^{Q/2}$	$2^{Q/2}K(2N_t^2+5N_t-2)+$ $2.2^{Q/2}$
LC-K-Best 1st itr	$4.2^{\frac{Q}{2}} + 2^Q +$ $K\left(2N_t^2 + 7N_t - 7\right) +$ $4K\left(N_t - 1\right)\left(1 - 2^{\frac{-Q}{2}}\right)$	$2^{\frac{Q}{2}} + 3.2^Q +$ $K\left(2N_t^2 + 5N_t - 5\right) +$ $4K\left(N_t - 1\right)\left(1 - 2^{\frac{-Q}{2}}\right)$
LC-K-Best i^{th} itr	$4.2^{\frac{Q}{2}} + 2^Q +$ $K\left(2N_t^2 + 11N_t - 11\right) +$ $4K\left(N_t - 1\right)\left(1 - 2^{\frac{-Q}{2}}\right)$	$2^{\frac{Q}{2}} + 3.2^Q +$ $K\left(2N_t^2 + 9N_t - 9\right) +$ $4K\left(N_t - 1\right)\left(1 - 2^{\frac{-Q}{2}}\right)$
LLR Computation	N_tQ	$2N_tQ$

4.6 Complexity results

This section provides the performance-complexity tradeoff results for iterative receiver based on MIMO detection algorithms in combination with LTE turbo decoder and LDPC decoder.
We restrict the complexity comparison with similar configurations used in the performance evaluation (c.f. section 3.9). Table 4.15 illustrates the main parameters and the configurations retained in the complexity evaluation.

Iterative MIMO detection complexity

The complexity of different detection algorithms in terms of number of operations for a 4×4 16-QAM system are summarized in Figure 4.1 and Figure 4.2 for 1^{st} and i^{th} iteration, respectively. The MAP algorithm presents the highest complexity (4.7×10^6 MUL, 4.6×10^6 ADD). It is not represented in the graph, but it is used as a reference to view the reduction in the complexity of other algorithms compared to the optimal detector. The complexity of STS-SD depends largely on the SNR. Its average complexity is computed through simulations over all SNR range. The average number of arithmetic operations is 90% lower than MAP algorithm. However it still has a larger complexity than other algorithms.

Table 4.15: Simulation parameters for complexity evaluation.

MIMO system	4×4 Spatial multiplexing (SM)
Modulation 2^Q-QAM	16-QAM, 64-QAM Gray mapping
Detector	Single tree-search (STS-SD) LC-K-Best, $K = 16$ I-VBLAST, MMSE-IC, MMSE-IC1,LC-MMSE-IC
Channel decoder	LTE turbo code $(13, 15)_o$ $R_c = 1/2$, $3/4$ Block Length $K_b = 1,024$ or $2,048$ bits
	LDPC code (IEEE 802.11n) $R_c = 1/2$, $3/4$ Codeword Length $N_b = 1,944$ bits
Interleaver	Random, size $= 1,024$ or $2,048$ (turbo code case)
Configuration	Turbo code case: $I_{out} = 4$, $I_{in} = 2$ (Scheme. 1) $I_{out} = 2$, $I_{in} = 4$ (Scheme. 2) $I_{out} = 8$, $I_{in} = 1$ (MMSE-IC)
	LDPC code case: $I_{out} = 4$, $I_{in} = [3,4,6,7]$

The complexity of the equalizer comprises the complexity of soft mapping and soft demapping. The complexity of K-Best and LC-K-Best decoders includes the complexity of SQR decomposition at the first iteration and LLR computation. The complexity of classical K-Best decoder is about 50% higher than that of LC-K-Best decoder. LC-K-Best complexity is approximately 30% higher than that of the MMSE equalizer and 50% lower than that of I-VBLAST. I-VBLAST requires more complexity due to the matrix inversion for each detected symbol. For i^{th} iteration, MMSE-IC shows 56 % higher complexity than MMSE-IC1 due to the matrix inversion for each detected symbol. However, MMSE-IC2 has 62% lower complexity than MMSE-IC1 since MMSE-IC2 approximation does not contain a matrix inversion. Comparing the complexity of

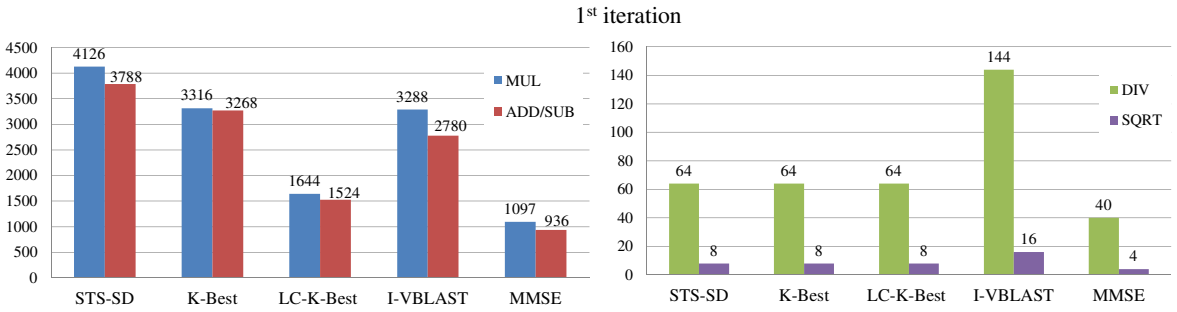


Figure 4.1: Complexity of a $\mathbf{4 \times 4}$ SM system with 16-QAM for different MIMO detection algorithms in terms of number of operations per symbol vector at the 1st iteration.

MMSE-IC1 equalizer and LC-K-Best decoder for the i^{th} iteration, we found that LC-K-Best decoder presents a lower number of arithmetic operations (22% MUL, 10% ADD). This is due to the fact that the equalizer repeats the matrix inversion for each iteration. However, LC-MMSE-IC algorithm proposed in [25] has a lower complexity than LC-K-Best decoder in terms of MUL (7%) and ADD(19%) with additional DIV and SQRT operations required by the matrix inversion. It is important to note that in LC-K-Best decoder, there is a number of comparisons to choose the best candidates that are not taken into consideration in the complexity comparisons. MMSE-IC2 presents the lowest complexity but with more performance degradation, it presents a reduction of 58% MUL and 52% ADD with a degradation of 1.5 dB compared to LC-K-Best decoder. Moreover, if the channel matrix is assumed to be quasi-stationary, the SQR decomposition as well as the matrix inversion associated with MMSE will be reused within the transmitted frame.

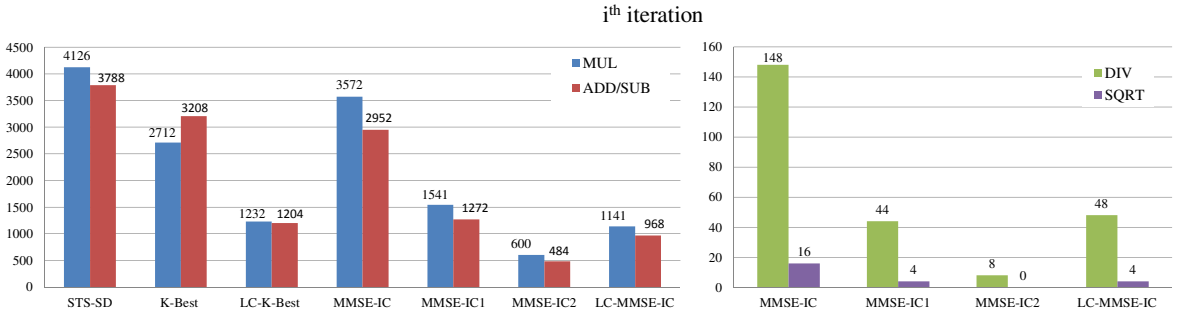


Figure 4.2: Complexity of a $\mathbf{4 \times 4}$ SM system with 16-QAM for different MIMO detection algorithms in terms of number of operations per symbol vector at the i^{th} iteration.

Figure 4.3 illustrates the complexity of different detection algorithms in terms of number of operations in the case of 4×4 spatial multiplexing systems using 64-QAM for the 1st and i^{th} iteration. We present only the complexity of LC-MMSE-IC since it has low complexity without performance degradation compared to MMSE-IC. Similarly, STS-SD presents more than 90% reduction in the complexity compared to MAP algorithm (1.2×10^9 MUL, 1.1×10^9 ADD). We note that the complexity of MMSE, LC-MMSE, I-VBLAST slightly increases because the complexity of soft mapper and soft demapper increases with the constellation size. Meanwhile, the complexity of computing filter coefficients will not be affected since the number of antennas is the same. We notice also that the complexity of LC-K-Best decoder is approximately twice as

much as that of LC-MMSE-IC equalizer. However, its complexity is about 40% lower than STS-SD (44% MUL and 45% ADD). It should be noted that even LC-MMSE-IC has a lower complexity, it presents performance degradation of more than 2 dB in the case of 64-QAM in Rayleigh channel, and more than 4 dB in realistic channels (c.f. section 3.9).

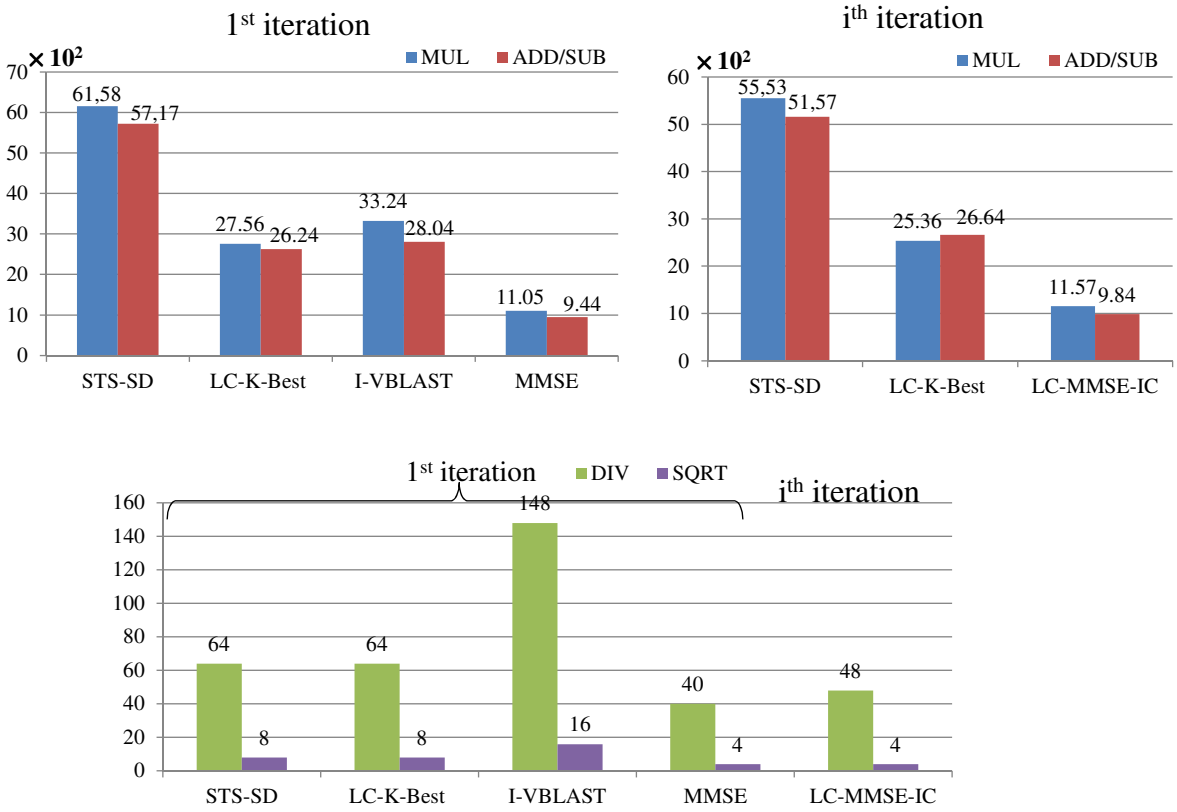


Figure 4.3: Complexity of a **4 × 4** SM system with **64-QAM** for different MIMO detection algorithms in terms of number of operations per symbol vector at the 1st and i^{th} iteration.

Complexity with turbo decoder

As we saw in section 3.9, the scheduling order and the number of iterations have a great impact on the system performance. The best trade-off schedule when performing I_{out} outer iterations, is to perform at least 8 turbo decoding iterations distributed equally into I_{in}. For this reason, we consider the two schemes having a difference of 0.5 dB at a BER level of 10^{-5} to view the impact of this performance degradation into the overall complexity. In the first scheme (scheme. 1), $I_{out} = 4$ and $I_{in} = 2$ iterations are performed. The second scheme (scheme. 2) has $I_{out} = 2$ and $I_{in} = 4$ as presented in Table 4.15. These two schemes present an equal number of turbo decoding iterations. Therefore, the complexity in terms of number of operations for the turbo decoder will be the same. However, the access to the memory will be changed.

Figure 4.4 summarizes the overall complexity of the receiver for one transmitted frame with these two schemes using different detection algorithms. As shown in Figure 4.4, a significant reduction of MUL operations between 40 $\sim$ 60% is obtained with scheme 2, and a reduction of ADD/SUB operations between 5 $\sim$ 25% is observed.

However, the number of Max operations remains the same since the same number of turbo decoding iterations is used. DIV and SQRT operations are more reduced in the case of MMSE-IC, MMSE-IC1 and LC-MMSE-IC by about 50% $\sim$ 60% due to reduction of matrix inversions. MMSE-IC2 presents a reduction of DIV operations by 25%. Moreover, comparing the overall complexity in the same scheme, we show that the complexity of STS-SD is much higher than LC-K-Best decoder (65 $\sim$ 67% MUL, 21 $\sim$ 33% ADD) at the expense of only 0.4 $\sim$ 0.25 dB of performance improvement at BER level of 1×10^{-5}. K-Best decoder presents lower complexity than STS-SD but higher than LC-K-Best decoder. LC-K-Best decoder shows approximately the same complexity as MMSE-IC1 equalizer. In addition, LC-K-Best presents a reduced complexity than I-VBLAST (20 $\sim$ 35% MUL, 2 $\sim$ 5% ADD, $\sim$ 50% DIV, $\sim$ 50% SQRT). The reason is that I-VBLAST requires multiple matrix inversions at the first iteration. However, LC-MMSE-IC has a lower complexity than LC-K-Best decoder in terms of MUL (15 $\sim$ 22% less) and ADD (4 $\sim$ 6% less) but requires more DIV (20 $\sim$ 50% more) and SQRT ($\sim$ 50% more) operations. Furthermore, the complexity of MMSE-IC2 equalizer is much lower than LC-K-Best decoder (40 $\sim$ 45% MUL less, 7 $\sim$ 13% ADD less). However, MMSE-IC2 presents a significant degradation of 1 $\sim$ 1.25 dB at 1×10^{-5} compared to LC-K-Best decoder.

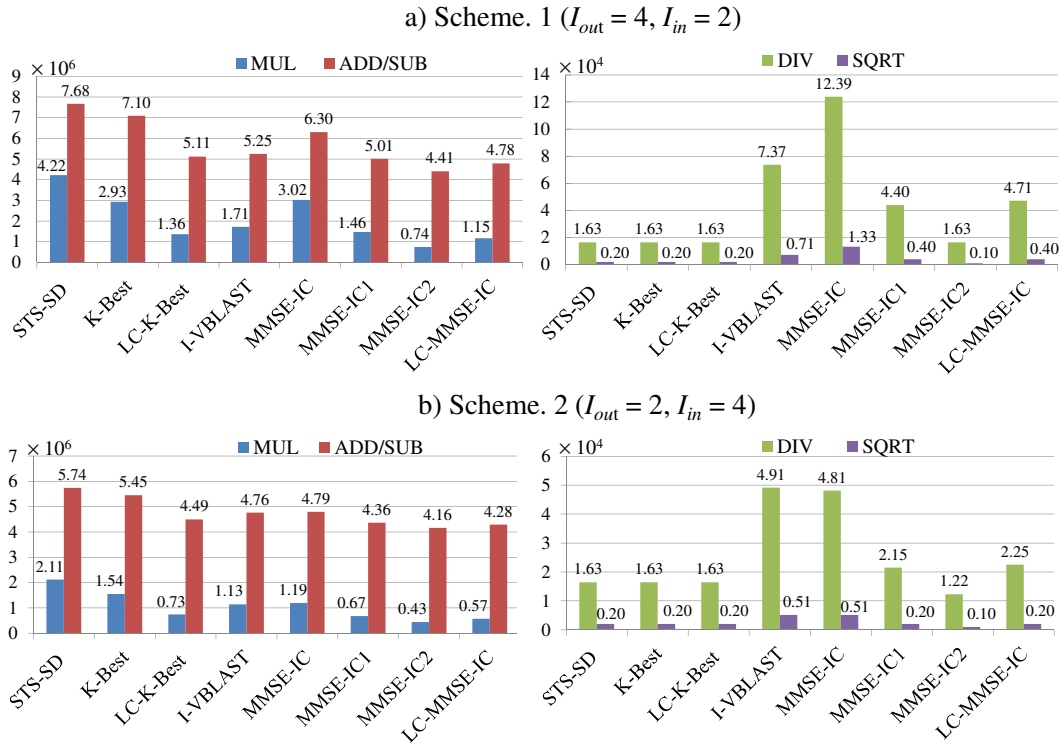


Figure 4.4: Overall computational complexity of a $\mathbf{4 \times 4}$ SM system with 16-QAM of different detection algorithms with two distinct system configurations, LTE turbo decoder $R_c = 1/2$, $K_b = 2,048$.

It is therefore worthy to compare the overall complexity of different receivers with approximately the same performance. We consider different configurations for different detection algorithms given BER $\approx 1 \times 10^{-5}$ at $E_b/N_0 = 2$ dB (except for MMSE-IC1 and MMSE-IC2 that have a degradation of 0.25 dB and 1 dB, respectively). Figure 4.5 illustrates the obtained results.

As Figure 4.5 shows, LC-K-Best decoder, LC-MMSE-IC and MMSE-IC2 equalizers have the lowest computational complexity. However, MMSE-IC2 presents more performance degradation. We remind that LC-K-Best decoder requires many comparison operations that are not considered in the complexity. MMSE-IC (8-1) has higher complexity even higher than STS-SD, due to matrix inversion for each symbol and for each iteration. MMSE-IC (4-2) has lower complexity than MMSE-IC (8-1) (55% MUL, DIV, SQRT; 32 % ADD) with small degradation (0.1 $\sim$ 0.2 dB). Furthermore, MMSE-IC1 and I-VBLAST present higher complexity than LC-K-Best since the matrix inversion is repeated at each iteration, while in LC-K-Best decoder SQR decomposition is only done at the first iteration. Moreover, LC-MMSE-IC presents lower complexity than LC-K-Best as previously discussed in Figure 4.4 with more DIV and SQRT operations.

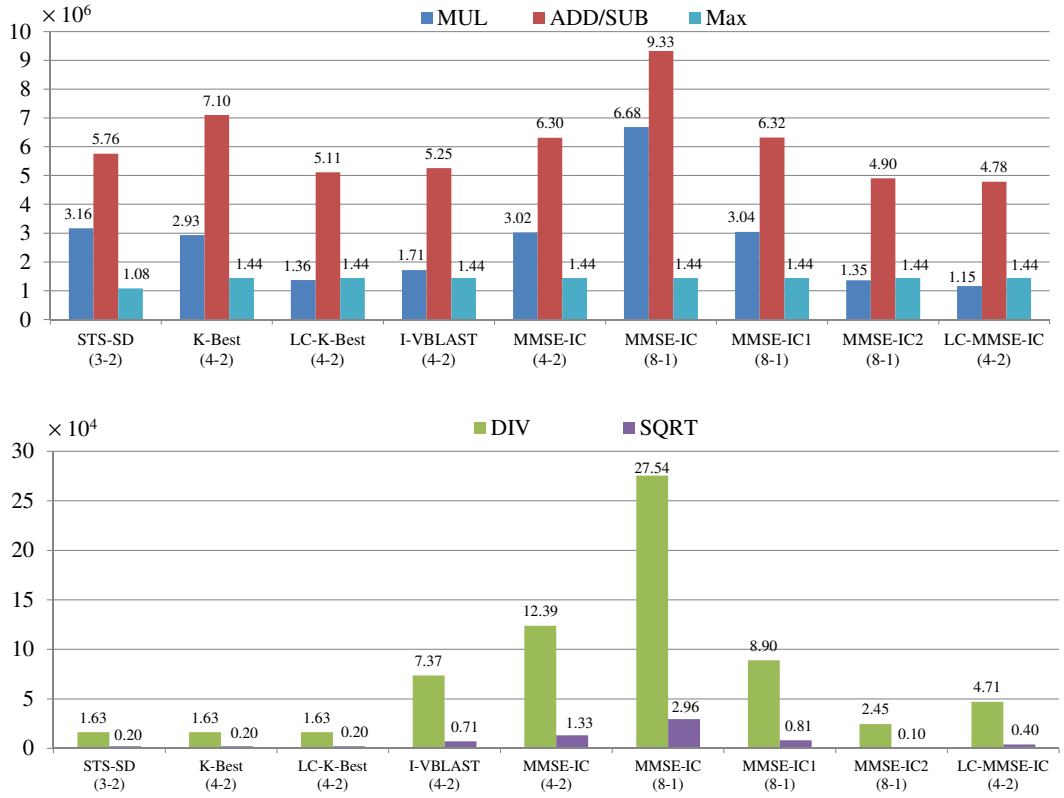


Figure 4.5: Overall computational complexity of a $\mathbf{4 \times 4}$ SM system with 16-QAM of different detection algorithms with BER $\approx 1 \times 10^{-5}$ at E_b/N_0= 2 dB, (I_{out}, I_{in}), LTE turbo decoder $R_c = 1/2$, $K_b = 2,048$.

Complexity with LDPC decoder

In this section, we compare the complexity of combining detection algorithms with LDPC decoder. For this comparison, we retain the most relevant detection algorithms namely STS-SD, K-Best, LC-K-Best and LC-MMSE-IC. We consider a 4×4 16-QAM system with $I_{\text{out}} = 4$ and a total of 20 iterations inside LDPC decoder. Figure 4.6 shows the total complexity of 4×4 16-QAM system of different detection algorithms with LDPC decoder for one transmitted frame. We show that complexity of K-best based receiver is approximately double of that of LC-K-Best receiver. LC-K-Best decoder

and LC-MMSE-IC equalizer present lower computational complexity than STS-SD. LC-MMSE-IC equalizer has also lower complexity than LC-K-Best decoder in terms of MUL(15%) and ADD(13%) with a larger number of DIV and SQRT.

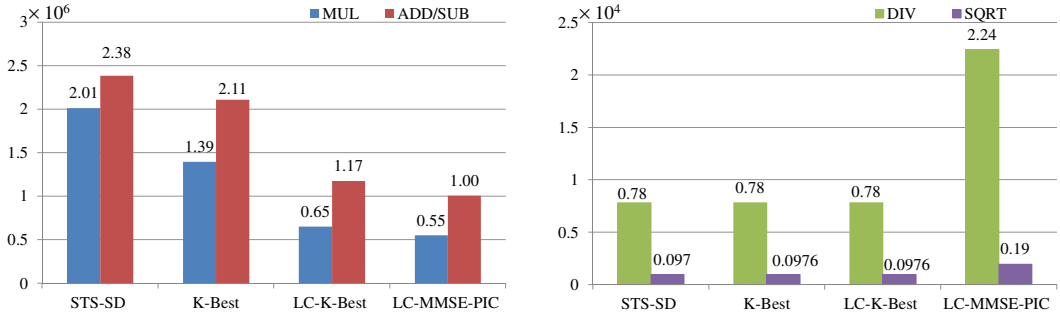


Figure 4.6: Overall computational complexity of a $\mathbf{4 \times 4}$ 16-QAM system of different detection algorithms with LDPC decoder $R_c = 1/2$, $N_b = 1,944$, $I_{out} = 4$ and $I_{in} = [3, 4, 6, 7]$.

Complexity comparison of iterative receivers with both coding schemes

We notice that the complexity of iterative receiver with LDPC decoder (Figure 4.6) is smaller than the complexity with turbo decoder (Figure 4.5). However, the complexity is evaluated with two different block lengths, the block length of turbo decoder is greater than LDPC decoder (approximately two times higher). For a fair comparison in terms of performance and complexity, similar block length have to be used. Therefore, we consider a block length $K_b = 1,024$ for turbo decoder and codeword length $N_b = 1,944$ in the case of LDPC decoder which gives a block length slightly lower (5%) than the turbo decoder case. The total number of iterations inside LDPC decoder and turbo decoder is chosen to be 20 and 8 iterations, respectively, because these numbers of iterations were found sufficient for the convergence of both decoders (c.f. section 3.8 and section 3.9). The number of operations consumed by LDPC decoder and turbo decoder per information block length with code rates $R = 1/2$ and $R = 3/4$ are listed in Table 4.16. We notice that LDPC decoder requires 20% to 40% less operations than turbo decoder. Note that the decoding complexity of turbo code is constant and does not depend on the code rate, because all code rates are generated from the mother coding rate $R = 1/3$. In contrast, the complexity of LDPC depends on the code rate. The decoding complexity decreases when the code rate increases.

Table 4.16: Complexity of turbo decoder (8 iterations) and LDPC decoder (20 iterations) in terms of the number of operations.

	Turbo (8 iterations)			LDPC (20 iterations)	
	ADD/SUB	Max	LUT	ADD/SUB	LUT
$R_c = 1/2$	1856 K_b	704 K_b	704 K_b	554 K_b	287 K_b
$R_c = 3/4$	1856 K_b	704 K_b	704 K_b	371 K_b	189 K_b

Figure 4.7 shows the computational complexity of the iterative receiver for one transmitted frame using both coding schemes and 16-QAM modulation. By comparing the

complexity of the receiver with both coding techniques, we notice that the complexity of iterative receiver with LDPC decoder is smaller than the complexity with turbo decoder in terms of addition, Max and LUTs. However, both receivers present approximately similar complexity in terms of MUL, DIV and SQRT.

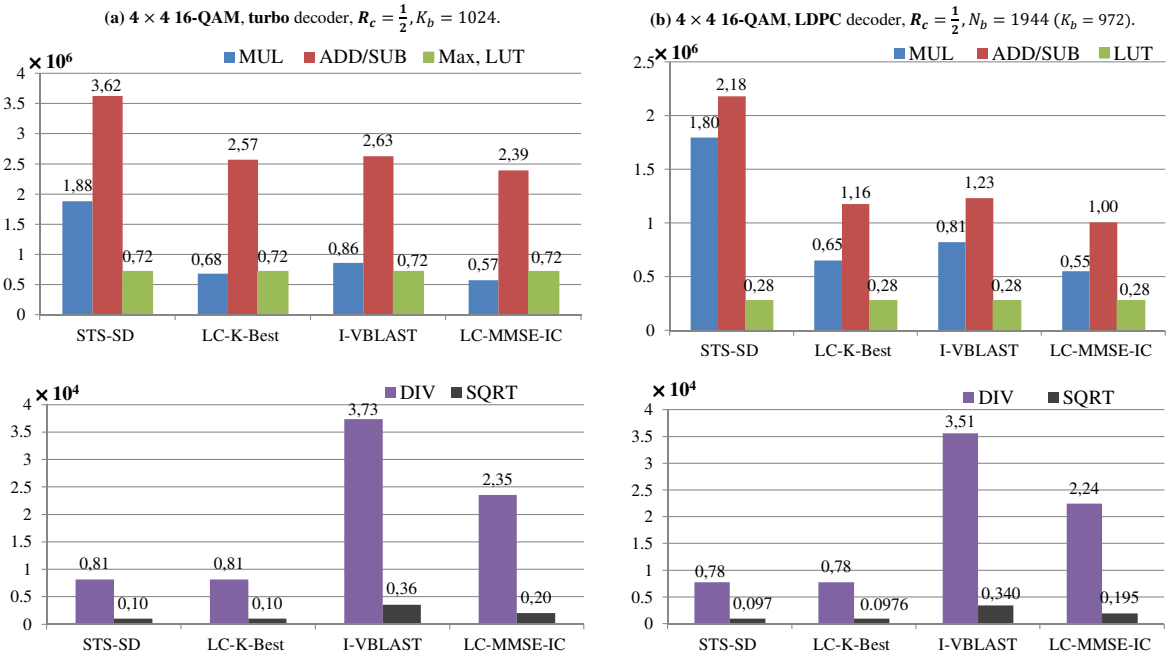


Figure 4.7: Complexity of a $\mathbf{4 \times 4}$ SM system with 16-QAM of different detection algorithms with **(a)** turbo decoder, and **(b)** LDPC decoder, $R_c = 1/2$.

It is therefore worthy to compare the complexity of the iterative receiver with high-order modulation and coding rate. Figure 4.8 illustrates the computational complexity of the iterative receiver for one transmitted frame in 4×4 spatial multiplexing system with 64-QAM. As shown in the figure, the complexity of the receiver based on STS-SD and LC-K-Best decoder increases significantly since the tree-search detection depends on the modulation order. The complexity of the receiver based on LC-MMSE-IC and I-VBLAST slightly increases compared to the case of 16-QAM due to the small increase in the complexity of soft mapper and soft demapper. Furthermore, the complexity of LC-MMSE-IC equalizer is much lower than LC-K-Best decoder ($\sim 55\%$ MUL, $\sim 26\%$ ADD). However, LC-MMSE-IC presents a significant degradation of about 2 dB in Rayleigh fading channel and more than 4 dB in realistic channels at BER level of 1×10^{-4} compared to LC-K-Best decoder (cf. section 3.9).

In addition, Figure 4.8b shows that iterative receiver with LDPC decoder presents low computational complexity in terms of ADD, LUTs. However, similar complexity of the receiver with both coding techniques is observed in terms of MUL, DIV and SQRT. Since MUL and DIV are more complex than ADD, Max and LUT, we can conclude that the complexity of iterative receiver with both coding schemes is comparable.

Conclusion

From this evaluation, we conclude that the performance and the complexity of the iterative receiver with turbo decoder and LDPC decoder are highly comparable. We

should also note that turbo decoder is recommended for small to moderate block lengths and coding rates. Meanwhile, LDPC decoder is more favored for large block sizes due to their superior performance and lower complexity. In addition, we see that LC-K-Best decoder achieves a good performance complexity trade-off compared to other detection algorithms. Furthermore, LC-K-Best decoder performs a breadth-first search that can be easily paralyzed and pipelined in hardware architecture as discussed in [18, 29]. LC-K-Best decoder can be also easily implemented and provide a high and fixed detection rates required in future communication systems.
Other aspects must be taken into consideration such as optimization of the computational complexity in hardware architecture, estimation of the required memory and the number of access to the memory since these aspects affect the scale of the receiver and the latency of the system. Moreover, the interleaver plays a major role to manage the access of the memory. These aspects can be the subject of future investigations.

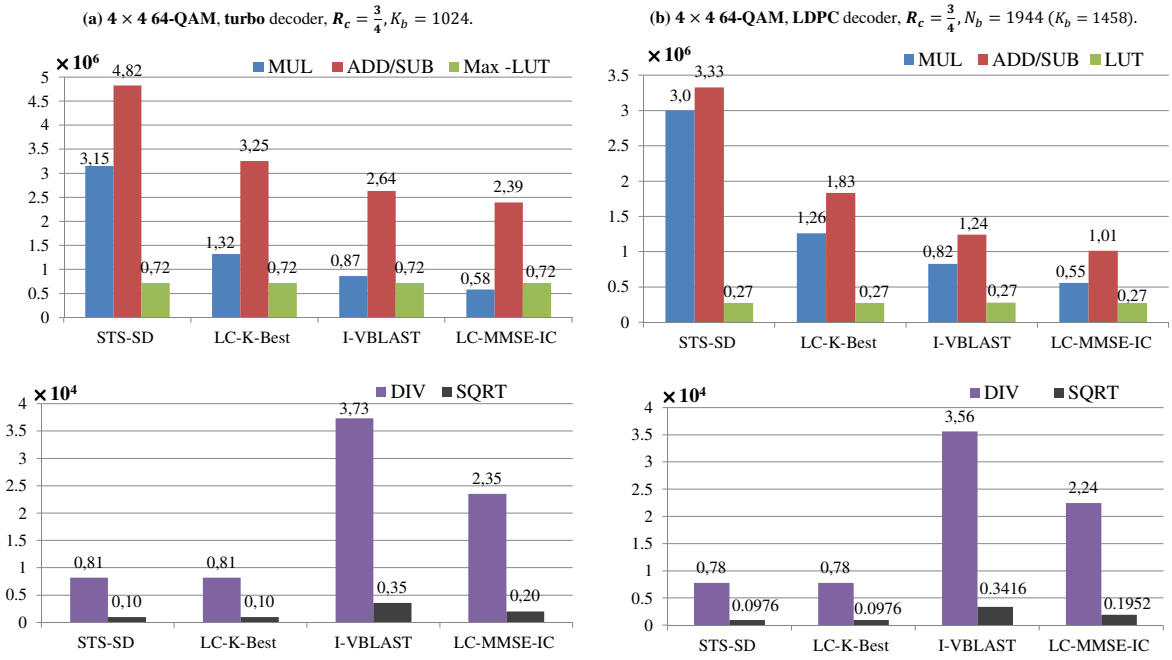


Figure 4.8: Complexity of a $\mathbf{4 \times 4}$ SM system with 64-QAM of different detection algorithms with **(a)** turbo decoder, and **(b)** LDPC decoder, $R_c = 3/4$.

4.7 Fixed-point representation of the iterative receiver

Theoretically, floating-point representation is commonly used to evaluate the performance of the system. Floating-point representation is characterized by a sign bit, a fraction (or mantissa) field, and an exponent field. The length of these fields is specified by a standard such as IEEE Standard 754-1985 for binary floating-point arithmetic. However, in practical implementation, fixed-point representation is more efficient in terms of area, power consumption and execution time. Fixed-point representation is characterized by a finite word length (wl), the location of the radix (binary) point, and whether the numbers are signed or unsigned. Herein signed fixed-point representation is considered.

In this section, we first describe the method adopted in order to find the required

word length of system parameters. We then evaluate the performance of the iterative receiver with different modulation orders and channel environments.

The conversion into a fixed-point format requires to specify a finite word length (wl) with a specific number of bits for integer parts ($iwl = n$) and for fractional parts ($fwl = f$). Any value x can be therefore expressed as:

$$\hat{x} = -2^n S + \sum_{i=-f}^{n-1} b_i 2^i, \quad -2^{n-1} \leq x \leq 2^{n-1} - 2^{-f}, \tag{4.6}$$

where S is the sign bit. The quantization step (resolution) q is given by $q = 2^{-f}$.

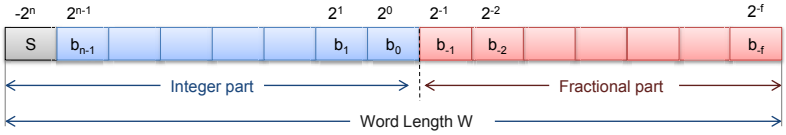


Figure 4.9: Fixed-point representation

Fixed-point format has two main challenges. First issue is the overflow problem due to the exceeding of the estimated dynamic range. Secondly, quantization errors may occur due to the limited precision. This quantization error is given by:

$$e = \hat{x} - x. \tag{4.7}$$

The value of the quantization error depends on the quantization process (rounding, truncation). For example in a rounding mode $e \in \left[-\frac{q}{2}, -\frac{q}{2}\right]$, and in truncation mode $e \in [0, q]$.

It is therefore necessary to define an appropriate fixed-point format with a trade-off between performance and complexity.

4.7.1 Fixed-Point conversion procedure

The conversion into a fixed-point format can be decomposed into two principal steps: determination of dynamic range and accuracy evaluation. Two distinct approaches can be used to evaluate the fixed-point representation of the system: simulation based approaches and analytical approaches. Simulation based approaches can be applied to all types of system. However, their main drawback is their high execution time. The analytical approaches try to find a mathematical expression and can only be applied for linear systems. In a non linear and unsmooth operations, the analytical approach is difficult to be evaluated.

Therefore, in our study, simulation based approaches are used to find the word length. The system is decomposed into many sub-blocks, then the required word length of each variable is determined in each sub-block. The different steps for the refinement of the algorithm into a fixed-point format are summarized in Figure 4.10.

First, the floating-point algorithm which is considered as a reference algorithm must be well adapted and configured according to system requirements. Mathematical approximations can be used to simplify the computational complexity of some expressions.

Next step consists in converting the algorithm into a suitable fixed-point arithmetic. This conversion requires to determine the format of each variable through the evaluation of its dynamic range and its precision, as well as the quantization of linear operations (add, multiply..) and non linear operations (division...) as shown in Table 4.17.

The dynamic range (iwl) of each variable is estimated by determining its maximum and minimum value through examining its histogram under excessive simulations to avoid overflow problem. Hence, an overestimation of the range may occur leading to the presence of non used bits and more implementation cost. Another approach consists in determining the probability density function of the variables and then estimating their dynamic ranges by accepting a certain overflow probability without significant degradation of system performance. The precision is determined by specify a minimum number of fractional bits (fwl) with an acceptable level of performance degradation.

In the last step, the performance of the system is evaluated and compared with the reference algorithm. In most communication system, BER is used as a metric for the performance evaluation. If the performance of the system is not acceptable, we increase the precision of each variable while keeping the other variable unchanged. We repeat this step until the desired performance is obtained.

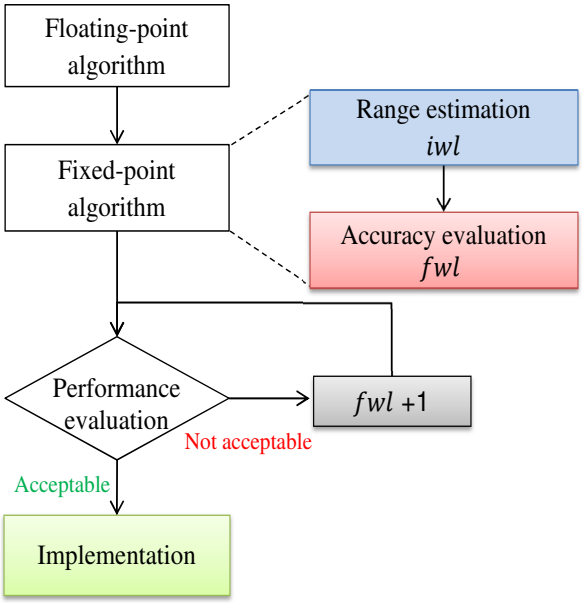


Figure 4.10: Fixed-point conversion process.

Once the number of bits of all variables is determined, mathematical operations between fixed-point numbers require an adaptation of the word length of the variables. For example, in the case of an addition, the binary points of both numbers must be aligned. However, the multiplication of two numbers results in more number of bits in both integer and fractional parts. The binary point position of the resulting number must be then determined by taking into account the binary point positions of these two numbers (Table 4.17).

We note that during the conversion of the algorithm into a fixed-point representation, quantization noises are introduced in the system due the difference between the

Table 4.17: Fixed-point arithmetic operations.

Operation	Variables format	Resulting format
Addition, Substation	$x1(n1, f1) \pm x2(n2, f2)$	$(\max(n1, n2) + 1, \max(f1, f2))$
Multiplication	$x1(n1, f1) \times x2(n2, f2)$	$(n1 + n2 + 1, f1 + f2)$
Division	$x1(n1, f1)/x2(n2, f2)$	$(n1 + f2 + 1, f1 + n2)$

value of the variable in infinite precision and its value in finite precision. These errors propagate into the whole system leading to significant performance degradation. In iterative system, the quantization noise is more critical since the noise propagate across iterations and may lead to the divergence of the system.

4.7.2 Fixed-point parameters

Long simulations have been conducted with different modulation orders and channel models to find the required word length for system variables. Similar simulation parameters of Table 4.15 are considered.

For MIMO detector, a quantization of the received signal, the channel coefficients, the noise variance and constellation symbols are required as illustrated in Figure 4.11. In the channel decoder, the quantization includes the received LLRs, the extrinsics LLRs and the internal computations and updates. In the case of turbo decoder, the computations include branch metrics, forward and backward state metrics. While, in the case of LDPC decoder, the check node messages and variable node messages updates have to be considered.

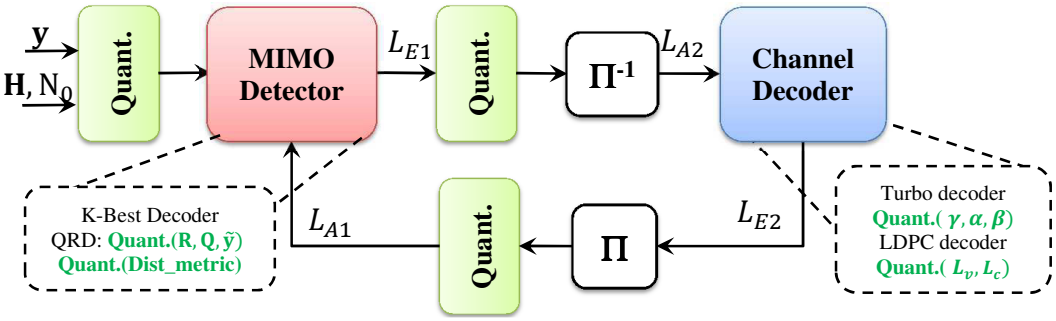


Figure 4.11: Fixed-point block diagram of the iterative receiver.

In our work, a quantization of channel decoder parameters is first carried out followed by a quantization of channel coefficients, received signal and detector parameters. Simulations in floating-point format are first performed with different modulation orders, coding schemes and channel models to provide an estimation of the range of all system variables. Then, we slightly change the dynamic range and we simultaneously increase the precision of each variable to find a suitable fixed-point format of the receiver. The fixed-point format will be represented with the notation (iwl, fwl), where iwl and fwl denotes the number of bits for integer part (including the bit sign) and fractional part, respectively.

Figure 4.12 shows the impact of the quantization of turbo decoder parameters on

BER performance. We show that 4 bits for integer part for LLR presents no performance degradation compared to floating-point curves. Meanwhile, a precision of 2 bits for fractional part is sufficient to achieve close to floating-point performance. However, in the case of high-order modulation, a precision of 3 bits is more suitable to avoid the small degradation on the performance. Therefore, the quantization of LLR to (4,3), $\bar{\gamma}(5,3)$, $\bar{\alpha}(6,3)$, $\bar{\beta}(6,3)$ is retained for the rest of this work.

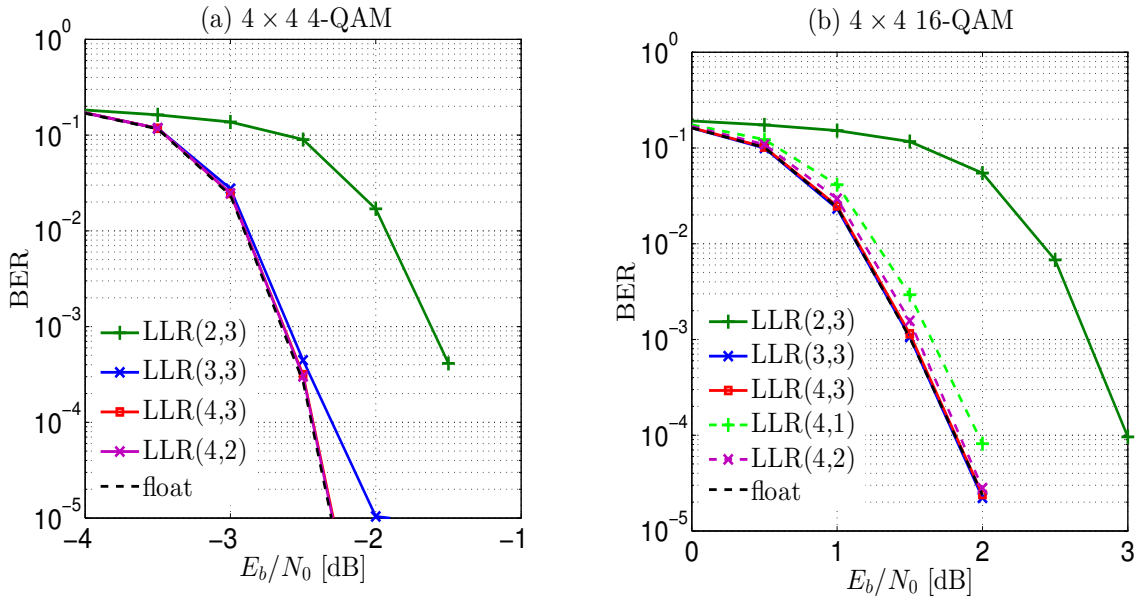


Figure 4.12: BER performance of a **4 × 4** coded MIMO system with LC-K-Best decoder in function of the quantization of turbo decoder parameters: **(a)** 4-QAM, **(b)** 16-QAM, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 1/2$ and $K_b = 2,048$ is used.

In Figure 4.13, the impact of the quantization of channel coefficients and noise variance on the performance of the iterative receiver for 4 × 4 16-QAM is presented. We notice that a precision value of 7 bits is sufficient to achieve close to floating-point performance.

The effect of quantization of the received signal is shown in Figure 4.14 for 4 × 4 16-QAM. We show that a precision of more than 5 bits with a dynamic range of 4 bits is required to achieve close to floating-point performance.

Table 4.18 summarizes the retained fixed-point representation of system parameters.

4.7.3 Simulation results

In this section, we evaluate the performance of the retained fixed-point format with different modulation orders and coding schemes. The simulations are based on a 4 × 4 spatially multiplexed MIMO-OFDM system. Rayleigh channel model is first used, then real channel models (EPA, EVA, ETU) are considered in order to evaluate the performance of the fixed-point arithmetic in more realistic scenarios. Table 4.19 lists the principle parameters of the simulations including the OFDM parameters.

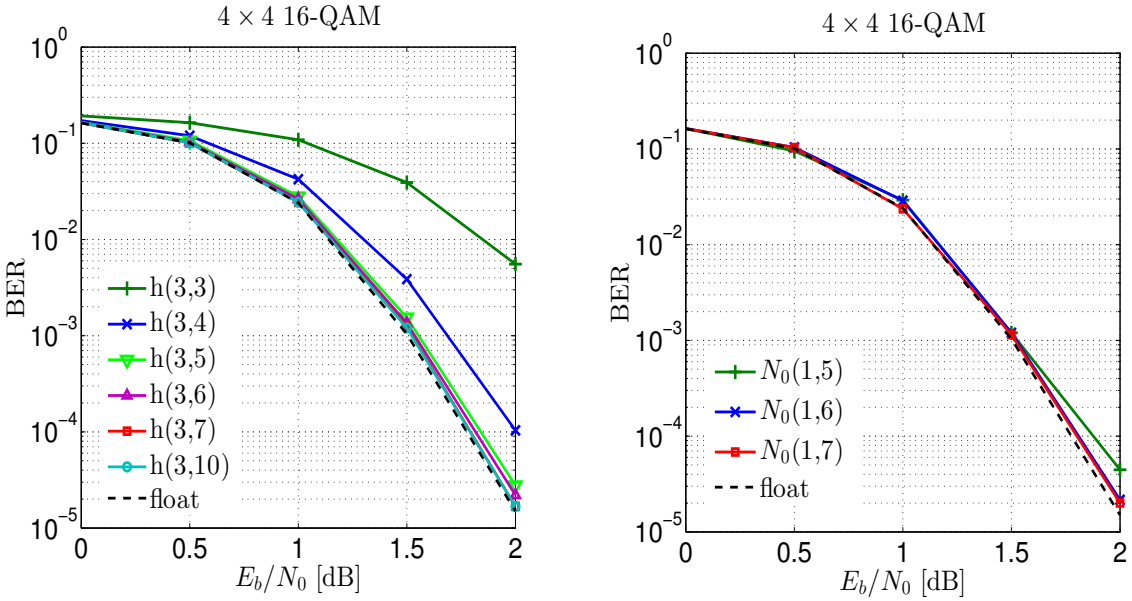


Figure 4.13: BER performance of a **4 × 4** coded MIMO system with LC-K-Best decoder in function of the quantization of **(a)** Channel coefficients and **(b)** Noise variance, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 1/2$ and $K_b = 2,048$ is used.

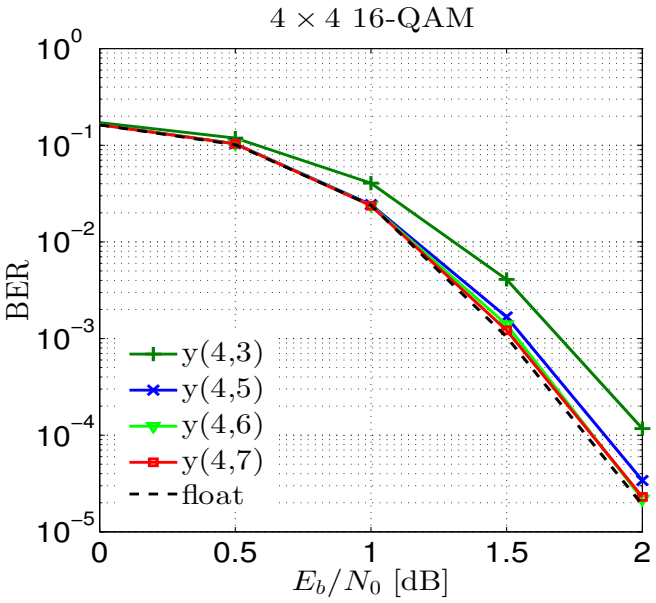


Figure 4.14: BER performance of a **4 × 4** coded MIMO system with LC-K-Best decoder in function of the quantization of the received signal vector, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 1/2$ and $K_b = 2,048$ is used.

Figure 4.15 depicts BER results of the fixed-point receiver in case of Rayleigh fading channels with 4-QAM and 16-QAM. As a reference, the floating-point curves of STS-SD and LC-K-Best decoder are plotted. The simulations show that fixed-point algorithm performs very close to the floating algorithm in the case of 4-QAM and 16-QAM. A performance loss of 0.2 dB at 10^{-5} is observed compared to STS-SD. However, in the case of 64-QAM with a coding rate $R_c = 3/4$ as shown in Figure 4.16, the retained

Table 4.18: Fixed-point representation of system variables.

	Parameters	(iwl, fwl)
Constellation	symbol $s(s_{\mathrm{Re}}, s_{\mathrm{Im}})$	(2,6),(2,9)
Channel	Channel coefficients h_{ij}	(3,7)
	Noise variance N_0	(1,7)-(1,10)
Turbo decoder	LLR	(4,3)
	$\bar{\gamma}$	(5,3)
	$\bar{\alpha}$, $\bar{\beta}$	(6,3)
K-Best decoder	$\mathbf{y}$	(4,7)
	Dist d_i	U*(5,7), U(5,9)
	$\mathbf{Q}$	(3,7), (3,9)
	$\mathbf{R}$	(4,7), (4,9)
	$\tilde{\mathbf{y}}$	(4,7), (4,9)
	LLR	(4,3)

* U refer to unsigned fixed-point arithmetic.

Table 4.19: Simulation Parameters.

MIMO system	4×4 Spatial multiplexing
Modulation 2^Q-QAM	(4, 16, 64)-QAM Gray mapping
Channel type	Flat Rayleigh fading EPA, EVA, ETU
Number of sub-carriers $N(N_c)$ Cyclic Prefix (CP) Bandwidth Carrier frequency f_c	1024 samples (600 used) Normal 5.2μs-4.7 μs 10 MHz 2.4GHz
Detector	single tree-search (STS) LC-K-Best decoder ($K = 8, 16, 32$)
Channel decoder	LTE turbo code K=4 $[13, 15]_o$ $R_c = 1/2$, 3/4 Block Length $K_b = 2,048$ bits
	LDPC code (IEEE 802.11n) $R_c = 1/2$, 3/4 Codeword Length $N_b = 1,944$ bits
Interleaver	Random, size = 2,048 (Turbo)
Inner iteration Outer iteration	$I_{in} = 2$ (turbo) $I_{in} = [3,4,6,7]$ (LDPC) $I_{out} = 4$

quantization parameters entail a BER performance loss of 0.25 dB at a BER level of 2×10^{-5}. We show that this gap is reduced by increasing the accuracy of the accumulated distance metric. A precision of 9 bits is sufficient to achieve close to

floating-point performance in this case. An accuracy of 9 bits for distance metrics and QRD are therefore adopted to avoid performance loss.

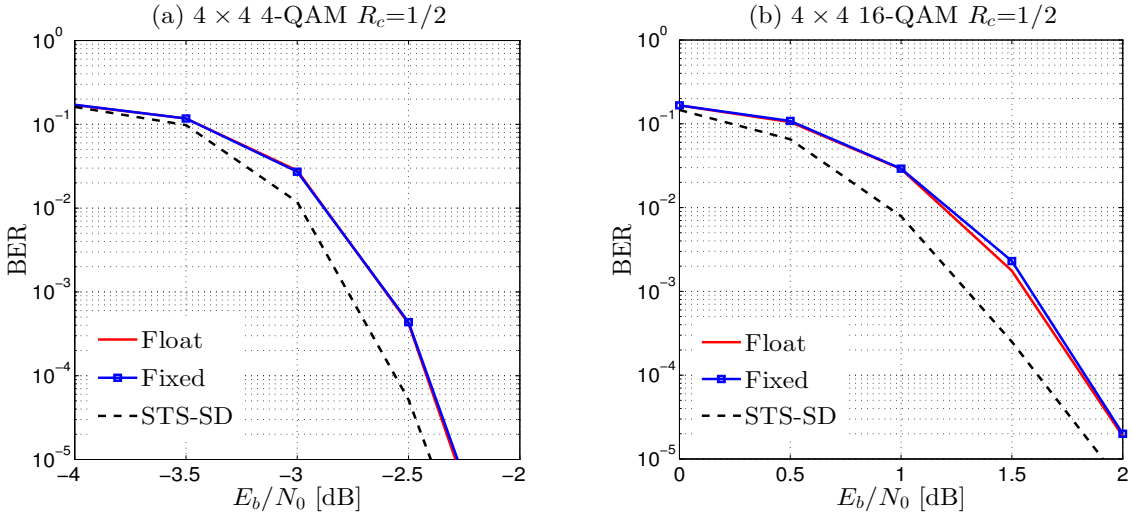


Figure 4.15: BER performance of fixed-point LC-K-Best based receiver for a **4 × 4** coded MIMO system on **Rayleigh** channel using **(a)** 4-QAM and **(b)** 16-QAM, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 1/2$ and $K_b = 2,048$ is used.

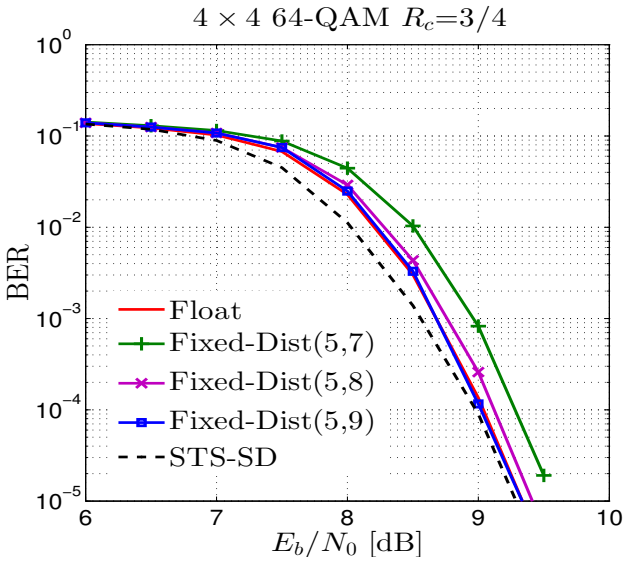
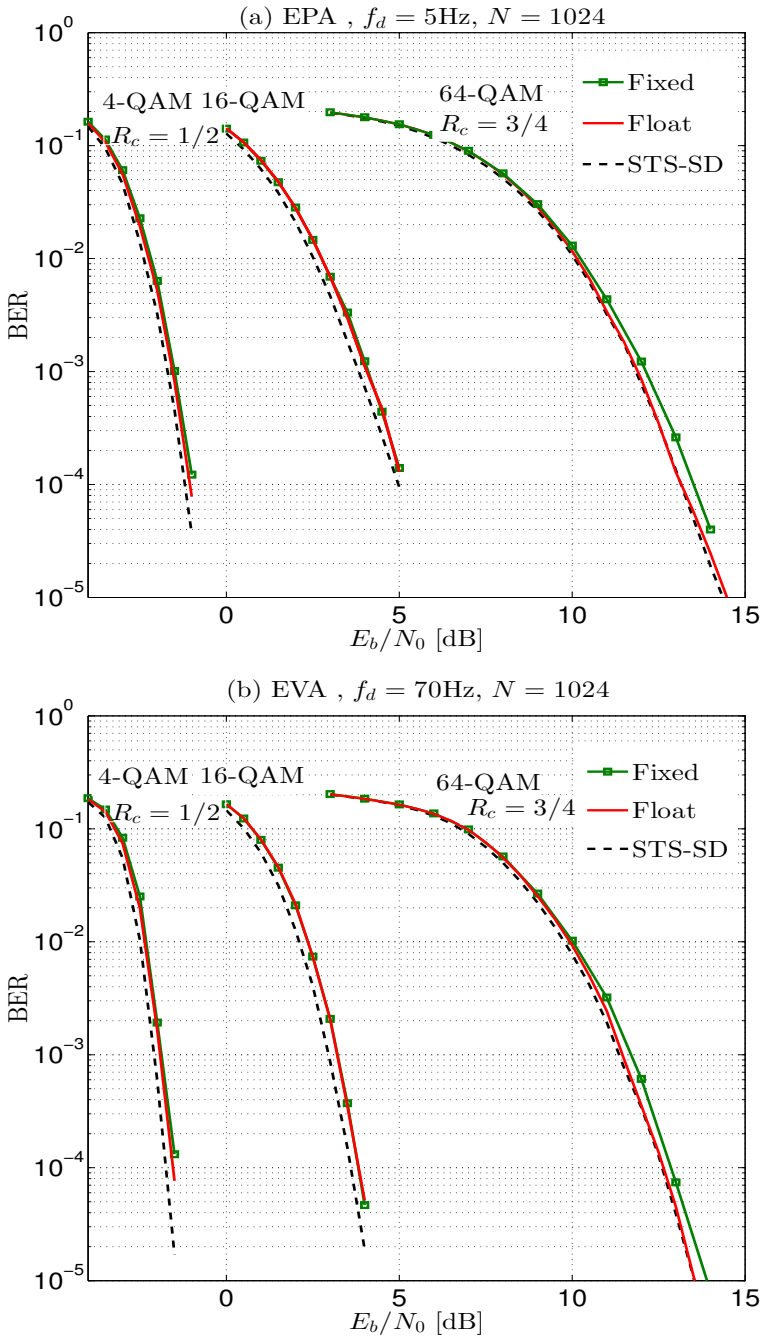


Figure 4.16: BER performance of fixed-point LC-K-Best based receiver for a **4 × 4** coded MIMO system on **Rayleigh** channel using 64-QAM, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 3/4$ and $K_b = 2,048$ is used.

Figure 4.17 shows BER performance of the fixed-point receiver in the case of real channel models (EPA, EVA and ETU). The results show that fixed-point representation presents similar performance as the reference floating-point curves in the case 4-QAM and 16-QAM. However, there is a performance loss below 0.2 dB in the case of 64-QAM at a BER level of 1×10^{-4}. We can see that STS-SD slightly outperforms LC-K-Best

decoder by about 0.2 dB with 4-QAM, 16-QAM and 0.1 dB with 64-QAM at a BER level of 1×10^{-4}. Hence our fixed-point arithmetic is able to achieve close to floating-point performance.

Figure 4.18 shows BER performance of the fixed-point receiver with LDPC decoder in Rayleigh fading channels. We show also that fixed-point curves achieves almost the same performance as floating-point curves with a degradation less than 0.1 dB in the case of 16-QAM and less than 0.2 dB in the case of 64-QAM.



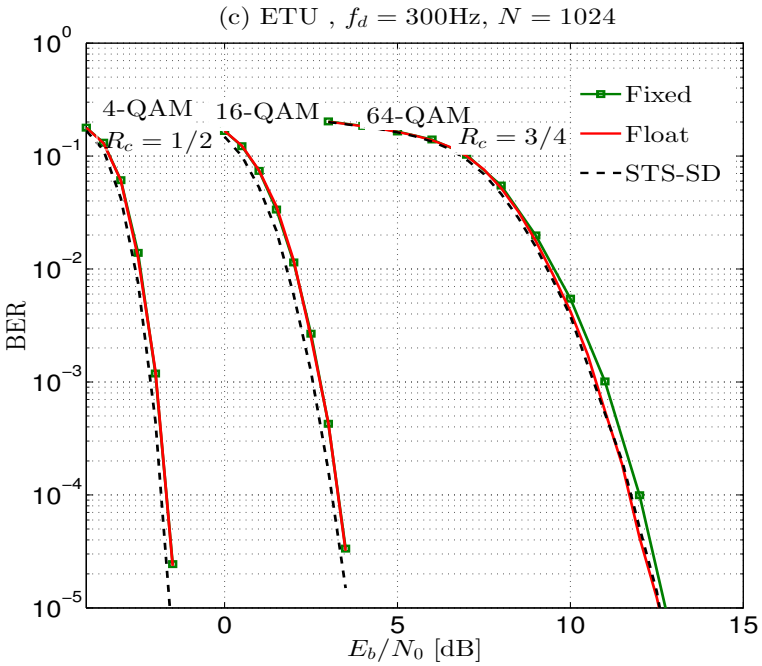
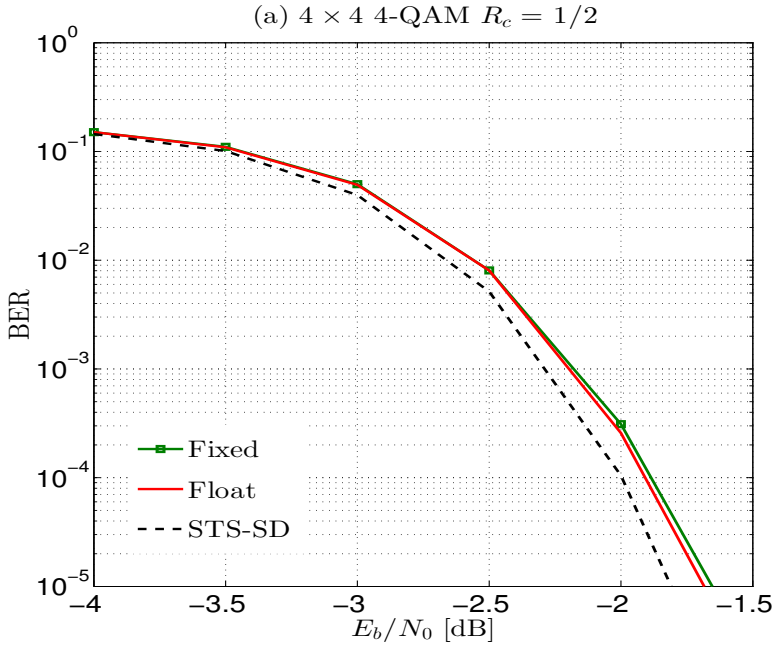


Figure 4.17: BER performance of fixed-point LC-K-Best based receiver for a **4 × 4** coded MIMO system on **(a)** EPA,**(b)** EVA and **(c)** ETU channels, $I_{out} = 4$, $I_{in} = 2$. Turbo decoder with $R_c = 1/2$ (4-QAM,16-QAM),$R_c = 3/4$ (64-QAM) and $K_b = 2,048$ is used.



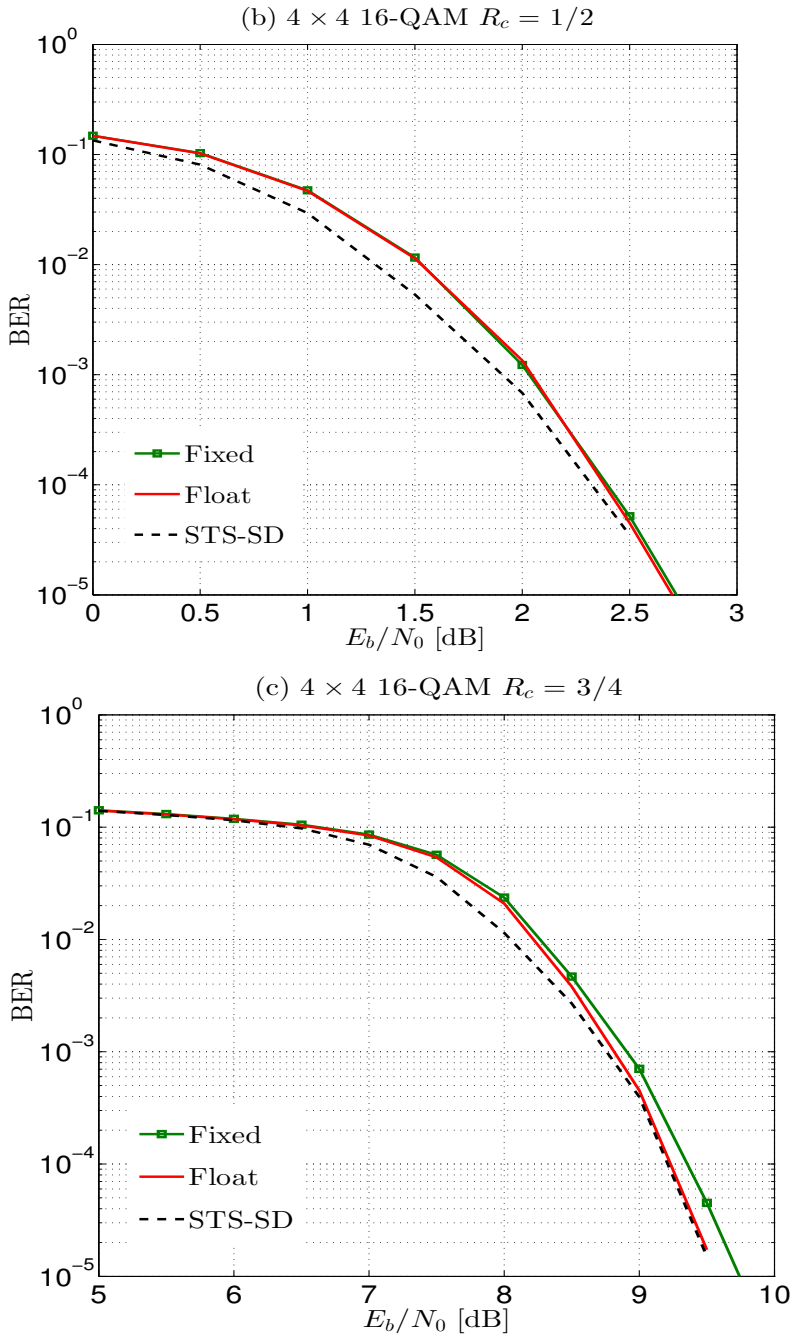


Figure 4.18: BER performance of fixed-point LC-K-Best based receiver for a 4×4 coded MIMO system on **Rayleigh** channel using (a) 4-QAM, (b) 16-QAM and (c) 64-QAM, $I_{out} = 4$, $I_{in} = [3, 4, 6, 7]$. LDPC decoder with $R_c = 1/2$ (4-QAM,16-QAM),$R_c = 3/4$ (64-QAM) and $N_b = 1,944$ is used.

4.8 Conclusion

In this chapter, a detailed study of the theoretical complexity of channel decoders and MIMO detectors has been presented in terms of number and type of real-valued operations. This computational complexity allows to estimate the potential cost of the algorithms. We compared their complexity with different system configurations. Simulation results show that the new schedule gives similar performance to the original schedule while saving a large amount of turbo decoder complexity and latency. Additionally, complexity results show that LC-K-Best decoder achieves a best trade-off between performance and complexity among the studied detectors.

In addition, we have presented an efficient fixed-point representation of the iterative receiver based on K-Best decoder. The performance of fixed-point receiver has been compared with floating-point performance using different modulation orders and channel models. Simulation results show that fixed-point representation achieves similar performance as floating-point system.

Future work can include other aspects like optimization of the computational complexity in hardware architecture, estimation of the required memory, parallel and pipeline implementation in real environments.

Chapter 5

Towards a real MIMO-OFDM systems

Contents

In the previous chapters, we always assumed perfect channel information at the receiver. However in practical communication system, the receiver has no *a priori* knowledge of the channel and has to estimate it using the received data samples. Moreover, in MIMO-OFDM system, the channel estimation is a critical issue since multiple channels are required to be estimated for each sub-carrier and for each link between transmit and receive antennas. This chapter first presents several channel estimation techniques for OFDM-MIMO system in LTE-based network. Their advantages, drawbacks and performances are analyzed and compared.

Additionally, in real transmission, the transmitted signal reaches the receiver after a propagation delay. Neither the instant of transmission nor the propagation delay is known at the receiver. Therefore, time synchronization as well as frequency synchronization are required at the receiver. We will briefly introduce the synchronization for MIMO-OFDM system. Finally, a testbed of MIMO-OFDM system using WARP board is introduced in order to highlight the key challenges encountered with a practical realization. This allows studying the impact of RF module, the time synchronization and channel estimation on real-time environments.

5.1 Imperfect channel estimation model

The imperfect channel estimation can be modeled as follows:

$$\hat{\mathbf{H}} = \gamma \mathbf{H} + \sqrt{(1-\gamma^2)}\mathbf{E}, \tag{5.1}$$

where $\mathbf{H}$ is the true channel matrix, $\hat{\mathbf{H}}$ is the estimated channel matrix, and $\mathbf{E}$ denotes the channel estimation error. The elements of $\mathbf{E}$ is assumed to be zero mean, unit variance and complex Gaussian. γ is an accuracy coefficient which indicates the accuracy of the channel estimation. The value $\gamma = 1$ corresponds to perfect channel estimation.

The channel estimation can be performed either in the frequency or in the time domain. For frequency domain channel estimates, mean square error (MSE) is usually used as performance measure of channel estimates, and it is defined by:

$$\text{MSE} = \mathbb{E}\left\{ \left\| \mathbf{H} - \hat{\mathbf{H}} \right\|^2 \right\}. \tag{5.2}$$

BER performance can be also considered to evaluate the performance of MIMO-OFDM system.

5.2 Channel estimation techniques

In general, channel can be estimated using a preamble, pilot symbols or data symbols. Three different categories of channel estimation have been developed in the literature including pilot-based channel estimation, blind channel estimation and semi-blind technique with decision directed [33].

Pilot-based channel estimation (data-aided)

Pilot-based or data-aided (DA) channel estimation technique consists in transmitting a training sequence which is known at both transmitter and receiver sides. The training sequence can be either a preamble or known symbols denoted as pilot symbols or reference symbols. The receiver utilizes these known symbols and the corresponding received symbols to estimate the channel [36, 179, 180, 181].
This technique is simple to implement. The major drawback of this technique is the loss of spectral efficiency due to the insertion of pilot symbols. The number of pilot symbols must be hence optimized for better use of spectral efficiency. Several types of pilot arrangement can be used depending on the channel variations as will be discussed in the next section (cf. section 5.3).

Blind technique (Non data-aided)

Blind technique does not required pilot insertion. It is based on the statistical proprieties of the received signals. This technique allows a better use of spectral efficiency. However, it is difficult to be implemented and has a long convergence time. Moreover, it suffers from performance degradation in fast fading channels [8, 182, 183].

Semi-blind technique, decision-directed channel estimation (DDCE)

Semi-blind technique is a combination of two previous techniques. It uses both the pilot symbols and received data symbols to estimate the channel. The channel is hence estimated by using pilot symbols and the previously estimated symbols. This technique is sensible to the error of detected symbols. The detected symbols can be based either on hard or soft-decision [184, 185].

In the rest of our study, we will focus on the techniques based on pilot insertion since they are able to achieve a good trade-off between performance and complexity. Figure 5.1 depicts the block diagram of pilot-based channel estimation for MIMO-OFDM system. At the transmitter, a pseudo noise (PN) generator is used to generate the pilot symbols that are then inserted into the resource grid (in LTE system) before being transmitted. At the receiver, the pilot symbols are first extracted from the received symbols after OFDM demodulation. The channel is then estimated using pilot sub-carriers. An interpolation is performed to obtain the channel at all sub-carriers. In the case of decision-directed channel estimation, estimated data symbols can be used to improve the estimation of the channel coefficients using an iterative process (feedback-loop).

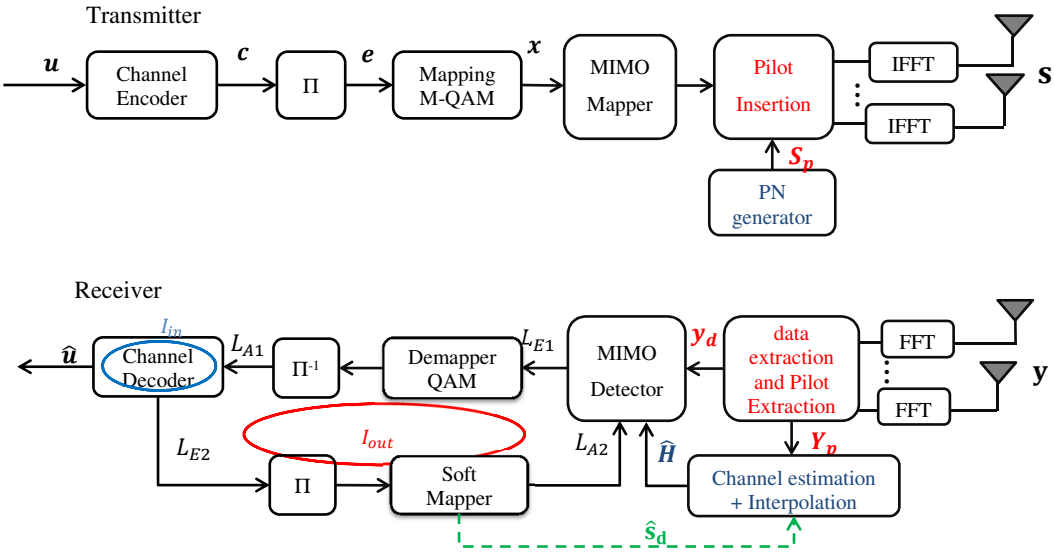


Figure 5.1: Block diagram of channel estimation for MIMO-OFDM system.

5.3 Pilot structures

In case of MIMO-OFDM system, pilot symbols can be inserted in frequency domain, in time domain or in both domains depending on channel characteristics and MIMO configurations. The insertion is basically done prior to OFDM modulation. In this section, channel constraints and MIMO constraints that must be considered into the pilot insertion are first presented followed by describing different pilot structures.

Channel constraints

The frequency and the time selectivities of the channel have a great impact on the system performance. To have a best estimate of the channel, the pilot must be

inserted in order to avoid the frequency and the time selectivities. The spacing of the pilot symbols in frequency domain depends on the coherence bandwidth of the channel which is related to the delay spread. Meanwhile in time domain, the pilot spacing depends on the coherence time which is related to the Doppler spread.

Let Δf_p the pilot spacing in the frequency domain, and Δt_p the pilot spacing in the time domain. In order to correctly estimate the channel, Δf_p and Δt_p must satisfy the following constraints:

$$\Delta f_p \leq B_c/2, \quad \Delta t_p \leq T_c/2, \tag{5.3}$$

where B_c and T_c are the bandwidth and time coherence of the channel, respectively.

Three different types of pilot structures are considered depending on the time and the frequency selectivities of the channel: block type, comb type and lattice type [180]. Figure 5.2 illustrates the structures of the pilot arrangement.

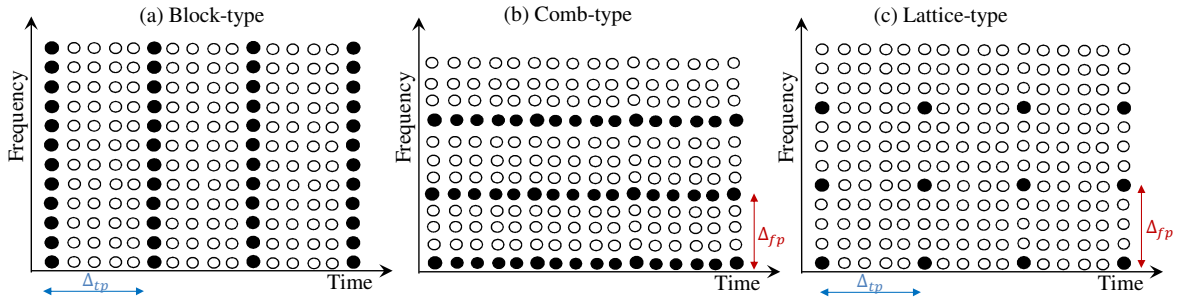


Figure 5.2: Pilot arrangement: **(a)** Block-type, **(b)** Comb-type, **(c)** Lattice-type.

Block type

Block structure consists in inserting the pilots in the frequency domain at all sub-carriers. Time domain interpolation is required to estimate the channel over all time symbols. This block type is robust in the case of frequency-selective channel and slow channel variation.

Comb type

Comb type is the inverse of block type, where the pilots are inserted in the time domain. These pilots are used for a frequency domain interpolation to estimate the channel over all sub-carriers. Comb type structure is well suitable for fast-fading channels.

Lattice type

In lattice structure, the pilot symbols are inserted along both the frequency and the time domain. Interpolation in both the time and the frequency domains must be therefore performed to estimate the unknown channels. The pilots can be distributed rectangularly, diagonally or even randomly. This structure can be robust against the frequency and the time selectivity of the channel depending on the pilot spacing.

In all cases, the number of pilot symbols compared to data symbols must be optimized in order to limit the loss of spectral efficiency. Many studies have been performed

to get optimum pilot location for a given channel with a best trade-off between spectral efficiency and channel estimates accuracy.
The allocation of power to pilot symbols with respect to data symbols is another important point in the channel estimation. In general, the powers of pilot and data symbols are equally distributed. However, increasing the power of pilot symbols compared to data symbols, known as *Boosted pilots*, leads to mitigate the effects of noise and improves the channel estimates accuracy.
Moreover, due to null sub-carriers at the edge of the spectrum in OFDM system, the estimation of channel coefficients at the edge may lead to a high performance degradation due to the *border effect* phenomenon further discussed in section (5.5.1).

MIMO constraints

The channel estimation in MIMO-OFDM system is a challenging task since the received signal is the superposition of signals from multiple transmit antennas [186, 187]. Indeed, each pilot symbol transmitted per antenna is received by N_r antennas. In order to avoid the interferences between pilot and data symbols, the pilot repartition must be orthogonal. In addition, the pilot symbols must be able to estimate the $N_r \times N_t$ links independently.

One method to accomplish this is to transmit pilot symbols prior to transmission, by turning off all transmit antennas at same instant time except for the i^{th} antenna, and sending a pilot symbol using the i^{th} antenna. Another way to estimate channel coefficients is to embed the pilots inside the transmitted symbols in an orthogonal way. Therefore the pilots used for an antenna are not used in the others antennas. In others meaning, the pilot symbols must be orthogonal in the frequency and in the time domains. The orthogonality can be achieved by the insertion of null symbols as used in LTE system. The main issue of this technique is the increase of the number of null symbols with the number of transmits antennas, which leads to further loss in spectral efficiency. Another way to achieve the orthogonality is to use an orthogonal sequence like Alamouti sequence or Hadamard sequence.

Figure 5.3 shows the principle of distribution of pilots, known as reference symbols, in LTE system for $N_t = 2$ and $N_t = 4$ antennas in the resource block (RB) [188]. The reference signals are inserted in the first and the third last element of the resource block in the time domain, Where as the reference signals are inserted every six sub-carriers in the frequency domain. The colored resource elements correspond to the reference signals transmitted by the antennas. Meanwhile, grey resource elements correspond to the null symbols. For example, resource elements used to transmit reference symbols on antenna 1 are not reused on antenna 2 for data transmission (Null symbols). This allows to estimate all channel links between the transmitter and the receiver without interference between antennas.

Spectral efficiency

The main drawback of pilot symbol insertion is the loss of spectral efficiency. This spectral efficiency loss can be expressed as:

$$\eta_{\text{loss}} = \frac{\text{number of pilot symbols}}{\text{number of data symbols in the frame}}. \tag{5.4}$$

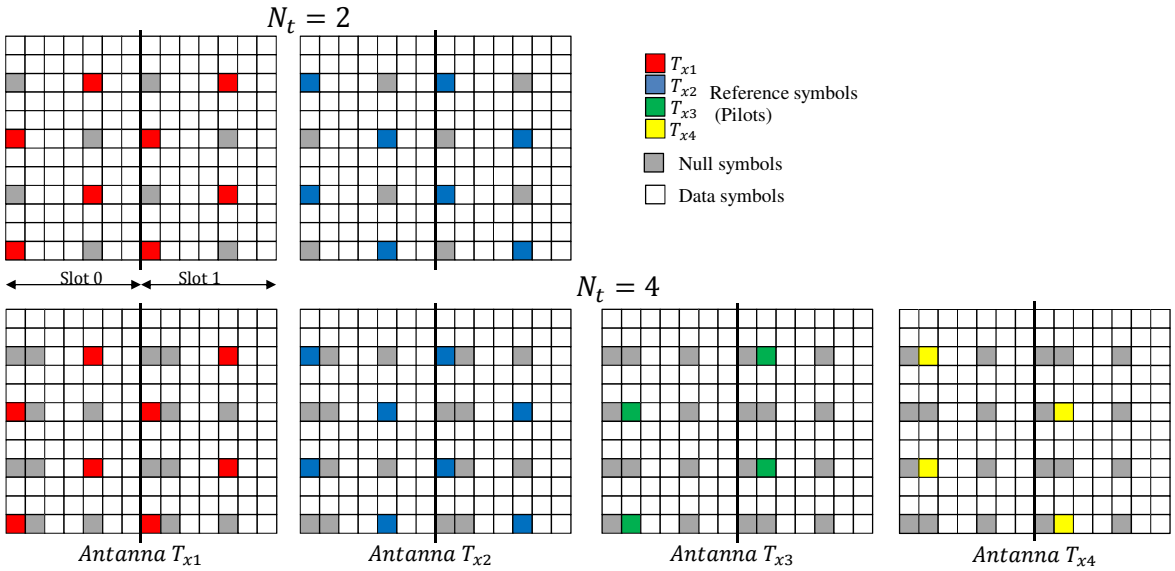


Figure 5.3: Reference symbols (pilots) in LTE system for $N_t = 2$ and $N_t = 4$ in the resource block (RB).

5.4 Pilot-based channel estimation techniques

Training symbols such as preamble or pilot symbols are usually used for channel estimation and provide generally good performance. In this section, the relevant pilot-based channel estimation techniques are reviewed namely least square (LS) and linear minimum mean square error (LMMSE) techniques [34, 35]. We note that channel estimation algorithms can be applied either in the frequency domain or in the time domain.

5.4.1 LS channel estimation

LS channel estimation is the simplest technique that does not require any knowledge of the channel statistics or noise variance. LS method finds the channel estimate $\hat{\mathbf{H}}$ that minimizes the square error between the received pilot symbols and the expected ones as follows [34, 35]:

$$\hat{\mathbf{H}}_p^{\mathrm{LS}} = \arg\min \|\mathbf{Y}_p - \mathbf{H}_p\mathbf{S}_p\|^2, \quad \mathbf{Y}_p = \mathbf{H}_p\mathbf{S}_p + \mathbf{W}_p, \tag{5.5}$$

where $\mathbf{S}_p$ denotes the pilot symbols. $\mathbf{Y}_p$, $\mathbf{H}_p$ and $\mathbf{W}_p$ are the received symbols, the diagonal channel matrix and the noise vector at pilot position, respectively. LS channel estimation is therefore given by:

$$\hat{\mathbf{H}}_p^{\mathrm{LS}} = (\mathbf{S}_p^H\mathbf{S}_p)^{-1}\mathbf{S}_p^H\mathbf{Y}_p. \tag{5.6}$$

For MIMO system, LS channel estimate from j^{th} transmit antenna to i^{th} receive antenna can be obtained by dividing the i^{th} received symbol at k^{th} pilot position $(y_{i,k})$ by the known pilot at j^{th} transmit antenna $(s_{j,k})$:

$$\hat{h}_{ij,k}^{\mathrm{LS}} = \frac{y_{i,k}}{s_{j,k}}, \quad k = 0, ..., N_p - 1. \tag{5.7}$$

These channel estimates at the pilot sub-carriers are used in the frequency domain for interpolation to get the channel coefficients at all sub-carriers.

The mean-square error (MSE) of the LS channel estimate is given by:

$$\begin{aligned} \mathrm{MSE}_{\mathrm{LS}} &= \mathbb{E}\left\{\left\|\mathbf{H}_p - \hat{\mathbf{H}}_p^{LS}\right\|^2\right\} = \mathbb{E}\left\{\left\|\mathbf{H}_p - \mathbf{S}_p^{-1}\mathbf{Y}_p\right\|^2\right\}, \\ &= \mathbb{E}\left\{(\mathbf{S}_p^{-1}\mathbf{W}_p)^H(\mathbf{S}_p^{-1}\mathbf{W}_p)\right\}, \\ &= \frac{\sigma_{w_p}^2}{\sigma_{s_p}^2}, \end{aligned} \tag{5.8}$$

where $\sigma_{w_p}^2$ and $\sigma_{s_p}^2$ are the variances of noise vector and transmitted vector at pilot position, respectively.
LS algorithm is simple to implement since it does not exploit the frequency and the time correlations of the channel. However it is very sensitive to noise and presents significant performance degradation compared to system with perfect channel estimation. In order to improve the channel estimation, noise variance and channel correlations are exploited by other algorithms.

Interpolation

The performance of LS channel estimator depends not only on the reliability of the channel at pilot position but also on the interpolation method used to find the channel coefficients over all transmitted symbols. The interpolation method must be robust against time and frequency selectivities of the channel. The interpolation can be applied in frequency and time domains [180]. The simplest interpolation methods are piece-wise constant and linear interpolations. The piece-wise constant interpolation assumes that the channel between pilots. However in linear interpolation, the channel coefficients at non-pilot sub-carriers are estimated through a straight line between pilot sub-carriers. Linear interpolation presents better performance than constant interpolation when the channel is more frequency selective. High-order polynomial interpolation like spline interpolation, Gaussian interpolation may be also used in time and frequency selective channels for better channel estimates accuracy at the expense of higher computational complexity.

5.4.2 LMMSE channel estimation

LMMSE channel estimation requires information about channel statistics and noise variance to improve the channel estimate accuracy. It is also referred to as 2D Winner filter that exploits the frequency and the time correlations [36, 189, 35]. The basic principle of LMMSE algorithm is to minimize the mean square error between the estimated channel and the real channel:

$$\hat{\mathbf{H}}^{\mathrm{LMMSE}} = \arg\min \mathbb{E}\left\{\left\|\mathbf{H} - \hat{\mathbf{H}}\right\|^2\right\}. \tag{5.9}$$

Using the orthogonality principle between the error vector and the channel, LMMSE-CE is represented by:

$$\hat{\mathbf{H}}^{\mathrm{LMMSE}} = \mathbf{R}_{HH_p}\left(\mathbf{R}_{H_pH_P} + \sigma_n^2\mathbf{I}_{N_p}\right)^{-1}\hat{\mathbf{H}}_P^{\mathrm{LS}}, \tag{5.10}$$

where $\mathbf{R}_{HH_p}$ denotes the cross-correlation matrix between all sub-carriers and pilot sub-carriers, $\mathbf{R}_{H_pH_P}$ is the auto-correlation matrix between pilot sub-carriers, and N_p is the number of pilot sub-carriers.

LMMSE is optimal in terms of MSE. However, it presents high computational complexity due to the matrix inversion and the requirements of correlation functions which are not known in practical systems. Low-complexity approximation of LMMSE estimator has been proposed in [190], where singular value decomposition (SVD) of the correlation matrix has been used to reduce the complexity.

Several methods have been investigated to improve the channel estimation with a lower computational complexity than LMMSE-CE namely transform domain channel estimation (cf. section 5.5).

Correlation functions of the channel

The elements of $\mathbf{R}_{HH_p}$ and $\mathbf{R}_{H_pH_P}$ can be computed as:

$$\mathbb{E}\left\{h_{k1,l1}h^*_{k2,l2}\right\} = r_f\left[k1-k2\right] r_t\left[l1-l2\right], \tag{5.11}$$

where k and l denote the sub-carrier (frequency) index and OFDM symbol (time) index, respectively. In an exponentially decreasing multipath power delay profile (PDP) channel, the frequency domain correlation $r_f[k]$ can be approximated by [191]:

$$r_f[k] = \frac{1}{1+j2\pi\tau_{rms}k\Delta f}, \tag{5.12}$$

where $\Delta f = 1/T_s$ is the sub-carrier spacing.
For a fading channel with maximum Doppler frequency f_m and Jake spectrum, the time domain correlation $r_t[l]$ can be computed as:

$$r_t[l] = J_0\left(2\pi f_m l T\right), \tag{5.13}$$

where $T = T_s + \Delta$, Δ is the guard interval time, and $J_0(x)$ is the zero-order Bessel function ($J_0(0) = 1$). We note that $r_t[l]$ decreases as the maximum Doppler frequency f_m increases.

5.5 Transform domain channel estimation (TD-CE)

The channel estimation method using transform domain has been proposed in the literature to reduce the impact of the noise and to improve the accuracy of LS channel coefficients. Two distinct TD-CE methods have been investigated: discrete Fourier transform (DFT) and discrete cosine transform (DCT) [37].
DFT-based method presents a good result in terms of noise reduction. Such a method works well when all the sub-carriers are modulated. However, in practical systems, null sub-carriers are placed at the edge of the spectrum as guard bands. These null sub-carriers may lead to performance degradation due to the *border effect* phenomenon.
Discrete cosine transform (DCT) has been hence suggested instead of DFT for mitigating the impact of *border effect*, owing to its capacity to reduce the high frequency components in the transform domain at the price of a weaker noise reduction [38].

However, when the number of null sub-carriers is important, DCT-based algorithm is not sufficient to reduce significantly the *border effect* [192]. To avoid this problem, several methods have been proposed in the literature namely truncated SVD [192, 193].

In TD-CE, CFR estimated by LS method is first converted into the transform domain using DFT or DCT. Then a smoothing filter is applied in a window, where the useful channel power is assumed to be concentrated within this window. After the smoothing process, the inverse of the transfer domain algorithm is applied to return back to the frequency domain. In the following, DFT-based method is only considered.

5.5.1 DFT-based channel estimation

Let $\hat{\mathbf{H}}_k^{LS}$ denotes the LS estimate of channel at k^{th} sub-carrier. This LS estimate can be converted into the time domain by applying an IFFT to $\left\{\hat{\mathbf{H}}^{LS}\right\}_{k=0}^{N-1}$:

$$\text{IFFT}\left\{\hat{\mathbf{H}}^{LS}\right\} = h[n] + z[n] = \hat{h}[n], \quad n = 0, 1, ..., N-1 \tag{5.14}$$

where $z[n]$ denotes the noise component in the time domain, and $h[n]$ is the CIR of the LS estimated channel without noise. A smoothing filter with a window W is applied to take only the most significant taps and ignore the other taps that contain noise as follows:

$$\hat{h}^{\text{DFT}}[n] = \begin{cases} h[n] + z[n] & n = 0, 1, ..., W-1 \quad W \geq L \\ 0 & \text{otherwise} \end{cases} \tag{5.15}$$

where W is the window size, and L is the number of channel taps ($W \geq L$).
$\hat{h}^{\text{DFT}}[n]$ is converted back to frequency domain by applying an FFT as follows:

$$\hat{\mathbf{H}}^{\text{DFT}} = \text{FFT}\left\{\hat{h}^{DFT}[n]\right\}. \tag{5.16}$$

Figure 5.4 shows the block diagram of DFT-based algorithm.

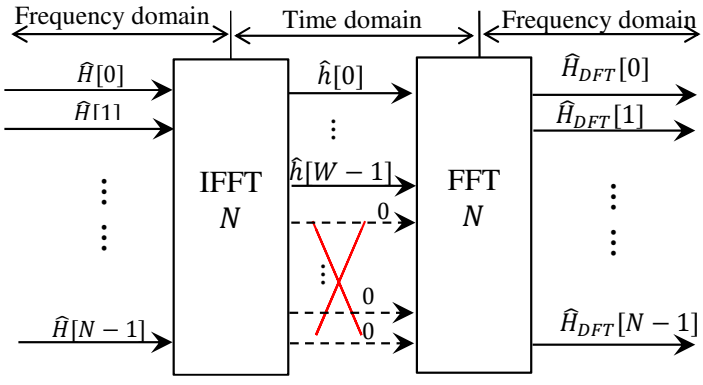


Figure 5.4: Block diagram of the DFT-based channel estimation ($W \geq L$).

The transform domain channel estimation process using DFT is represented in figure 5.5. IFFT is first applied to the channel frequency response. The transform domain is therefore the time domain. The time response is then reduced to retain only significant taps corresponding to the impulse response of the channel and zeroing others

terms. The frequency domain response is then obtained via FFT operation.
A window length of L represents the optimum window length that greatly reduces the noise effect. However, a practical window of cyclic prefix (CP) length is usually used since L is not known at the receiver [34, 37]. We note that the time domain processing can be considered as an interpolation technique in the case when $N_P < N$, since it exploits the time and the frequency correlations through the use of FFT/IFFT.

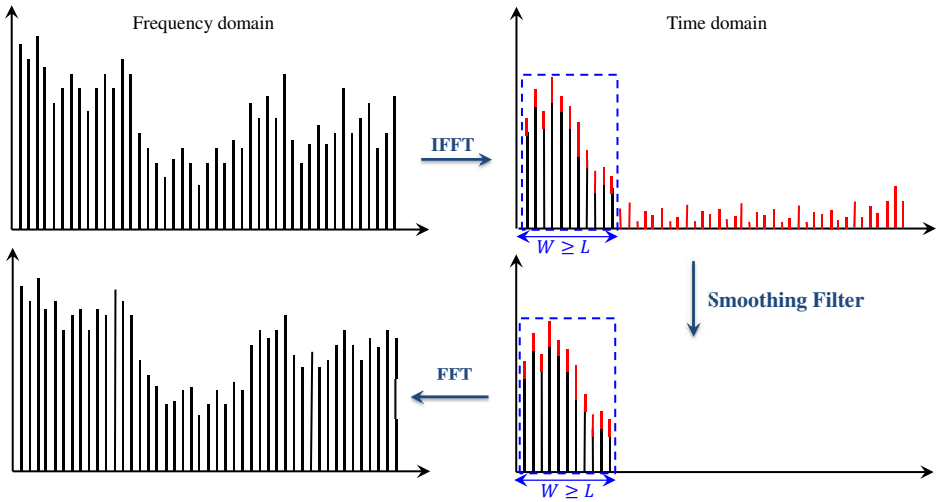


Figure 5.5: Transform domain channel estimation process using DFT.

DFT-based channel estimation can be used when all sub-carriers are modulated since it can able to reduce the noise effect and improve the performance of channel estimation [35]. However, in practical system, null sub-carriers are placed at the edge of spectrum to avoid interference with neighboring bands. This leads to discontinuity of the channel estimation at the edge of the spectrum, known as border-effect phenomenon [35, 187, 39]. In this case, the useful channel power is no longer concentrated on the first L taps. Therefore, the smoothing filter applied in the time domain will leads to loss a part of the channel power and significantly degrades the channel estimation as illustrated in Figure 5.6.

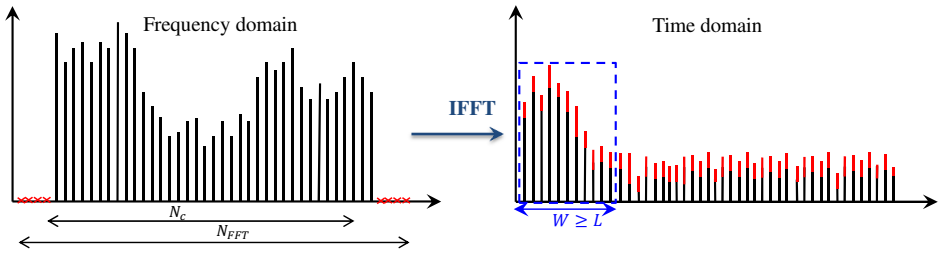


Figure 5.6: *Border effect* problem with DFT channel estimation when $N_c < N$.

5.5.2 Pseudo-inverse based channel estimation

In the previous section, we see that the main drawback of DFT-based channel estimation is the *border effect* phenomenon which degrades significantly the performance

and leads to an error floor. In [39], a robust channel estimation using the pseudo-inverse matrix has been proposed. The basic principle of the proposed approach is to minimize the distance between DFT-based channel estimates and the estimates:

$$\hat{h} = \arg\min_{h} \left\| \mathbf{F}_p h - \hat{\mathbf{H}}_P^{\mathrm{LS}} \right\|, \tag{5.17}$$

where $\mathbf{F}$ is the N-points DFT matrix, $\mathbf{F}_p$ denotes the matrix corresponding to the pilot sub-carriers and contains the first Δ columns of $\mathbf{F}$. The solution of this minimization problem is given by the pseudo-inverse matrix, also referred to as generalized inverse:

$$\hat{h}^{pinv} = \mathbf{F}_p^{\dagger} \hat{\mathbf{H}}_P^{\mathrm{LS}}, \tag{5.18}$$

where $\mathbf{F}_p^{\dagger} = (\mathbf{F}_p^H \mathbf{F}_p)^{-1} \mathbf{F}_p^H$ denotes the pseudo-inverse matrix of $\mathbf{F}_p$. We then return back to the frequency domain:

$$\mathbf{H}^{pinv} = \mathbf{F}_c \mathbf{F}_p^{\dagger} \hat{\mathbf{H}}_P^{\mathrm{LS}}. \tag{5.19}$$

Consequently, the channel estimation in the transform domain can be performed using the pseudo-inverse instead of the transpose conjugate. This approach can be considered as a generalization of the time domain transform. We note also that the pseudo-inverse matrix is constant, independent of the channel and can be pre-computed.

The pseudo-inverse can be computed by using matrix inversion. However, the efficient method to compute the pseudo-inverse is to use the singular value decomposition (SVD) [39]. SVD decomposes the matrix $\mathbf{F}_p$ into three matrices:

$$\mathbf{F}_p = \mathbf{U}_p \Sigma_p \mathbf{V}_p^H, \tag{5.20}$$

where $\mathbf{U}_p$ and $\mathbf{V}_p$ are unitary matrices, and Σ_p is a diagonal matrix with singular value on its diagonal.

$$\Sigma_p = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \sigma_p \end{bmatrix}. \tag{5.21}$$

The pseudo-inverse of the matrix $\mathbf{F}_p$ can be computed as:

$$\mathbf{F}_p^{\dagger} = \mathbf{V}_p \Sigma_p^{\dagger} \mathbf{U}_p^H. \tag{5.22}$$

The precision of the pseudo-inverse depends on the matrix $\mathbf{F}_p$. The condition number of the matrix $\mathbf{F}_p$ is a good indicator of the pseudo-inverse accuracy. This condition number is defined by the ratio between the greatest and the lowest singular value of $\mathbf{F}_p$:

$$cn(\mathbf{F}_p) = \frac{\sigma_1}{\sigma_p}. \tag{5.23}$$

If cn is small, the matrix is well-conditioned. However, if cn is high the matrix is ill-conditioned and the pseudo-inverse computation becomes less accurate.
When null sub-carriers are placed at the edge of the spectrum, the condition number becomes very high. In order to reduce the *border effect*, it is necessary to reduce the condition number.

5.5.3 Truncated SVD based channel estimation

Truncated SVD (TSVD) based channel estimation has been proposed to improve the accuracy of the channel by truncating the small singular values [194, 195, 193]. A threshold is used for this purpose. Any singular value smaller than the given threshold is set to zero. Figure 5.7 illustrates the principle of TSVD-based channel estimation.

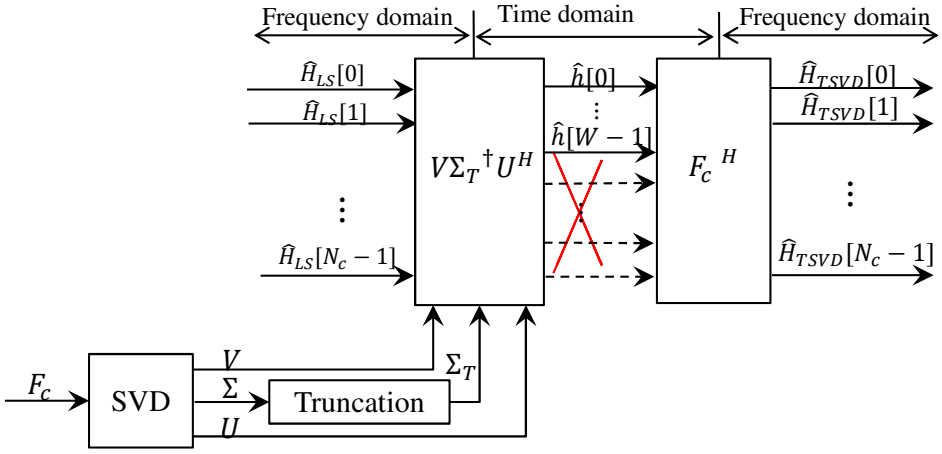


Figure 5.7: Block diagram of TSVD-based channel estimation.

Using a threshold value, the channel matrix can be then expressed as:

$$\hat{h}^{TSDV} = \mathbf{V}_p \Sigma_T^{\dagger} \mathbf{U}_p^H . \hat{\mathbf{H}}_P^{\mathrm{LS}}, \tag{5.24}$$

where Σ_T is obtained from Σ_p by setting the singular value smaller than the threshold to zero:

$$\Sigma_T = \begin{bmatrix} \sigma_1 & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \sigma_2 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \sigma_T & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 & 0 \end{bmatrix}. \tag{5.25}$$

An optimized threshold must be used in order to enhance the channel estimation accuracy. If the threshold value is set too high, the pseudo-inverse will be less accurate. Meanwhile, a smaller threshold value will lead to strong discontinuity. A typical threshold of 10% of the maximum singular value has been shown to give good performance [196]. In [40, 193], a technique for the determination of an optimum truncation threshold for any MIMO-OFDM system has been proposed.

Additionally, compressive sensing (CS) was recently proposed for channel estimation in 2008 [197]. CS utilizes the concept of randomness to recover sparse or compressive signals, i.e, signals that are only constituted of few essential samples or from a very limited number of measurements. For more details about this method, we refer to [197, 198] and references therein.

5.6 Advanced channel estimation techniques

5.6.1 Semi-blind channel estimation

Semi-blind channel estimation method uses pilot symbols to provide initial channel estimates. Then detection is carried out based on these initial channel estimates. The estimated symbols are therefore used to refine the channel estimation. In this section, we briefly review the common semi-blind techniques.

Decision-Directed channel estimation

In Decision-Directed channel estimation, once the data symbols have been estimated, they can be used to subsequently update the estimated channel coefficients [199, 200, 201]. Such a method is sensitive to error propagation, where any error in the detected symbol may be propagated and can significantly degraded the performance of the system.

EM-based channel estimation

Expectation-Maximization (EM) based channel estimation is an iterative technique for finding maximum likelihood (ML) estimates of a channel [202, 203]. EM algorithm consists of two iterative steps: expectation (E) step and maximization (M) step. The expectation step estimates the corresponding component of transmitted signal (complete data), Where as in the maximization step, the channel estimates are updated. EM algorithm does not require *a priori* knowledge of the channel fading statistics. Moreover, it is able to improve the estimation accuracy in fast fading scenarios. Despite the advantages of EM algorithm, the channel estimation of MIMO-OFDM system is not straightforward, because the computational complexity of EM algorithm increases exponentially with the number of transmitted signals and the size of the constellation. Furthermore, high number of iterations must be performed to converge to ML estimate. The space-alternating generalized expectation-maximization (SAGE) algorithm has been therefore proposed to accelerate the convergence of EM algorithm by sequentially updating the parameters instead of simultaneous updates [204, 203]. A joint channel estimation with detection and decoding for MIMO-OFDM system has been considered in [205]. There are also adaptive channel estimation schemes like least mean square (LMS), recursive LS (RLS) and Kalman filter [206, 187, 207].

5.6.2 Blind channel estimation

Blind channel estimation uses the statistical properties of received signals to estimate the channel without overhead of training signals [8, 182, 183]. However, it often needs a large number of received symbols to extract the statistical properties. Furthermore, its performance is usually worse than other conventional channel estimation techniques that employ the training sequence. It imposes also high complexity and slow convergence. Examples include constant modulus algorithm and the subspace-based channel estimation [183].

5.7 Performance results

In this section, we evaluate the performance of several pilot-based channel estimation techniques in LTE system environments. The complexity of these techniques will not be considered in this work. Perfect time and frequency synchronization are assumed. 2×2 and 4×4 SM systems are considered. The simulation parameters are listed in Table 5.1.

Table 5.1: Simulation Parameters.

MIMO system	2×2, 4×4 SM
Modulation 2^Q-QAM	4-QAM, 16-QAM Gray mapping
Channel type Number of sub-carriers $N(N_c)$ Cyclic Prefix (CP) Bandwidth	EPA, EVA, ETU 1024 samples (600 used) Normal $5.2\mu s$(80 samples)-$4.7\mu s$(72 samples) 10 MHz
Carrier frequency f_c	2.4 GHz
Detector	LC-K-Best decoder ($K = 8, 16$)
Channel decoder	LTE turbo code $(13, 15)_o$ $R_c = 1/2$ Block Length $K_b = 2,048$ bits
Inner iteration Outer iteration	$I_{in} = 2$ (turbo) $I_{out} = 4$

Three channel environments with different delay spreads and Doppler frequencies are considered in order to study the impact of frequency and time selectivities of the channel on the performance of the system. The channel characteristics of these environments including the coherence bandwidth and the coherence time are summarized in Table 5.2. The coherence bandwidth and the coherence time values are computed according to the equations (cf. section 1.2.3). The number in the parentheses correspond to the number of the sub-carriers (sc) and the number of symbols (symb) over which the channel can be assumed constant. For the performance evaluation, various channel estimators are considered as shown in Table 5.3.

Figure 5.8 shows MSE on different sub-carriers with classical DFT-based channel estimation method and TSVD method. We show that when all the sub-carriers are modulated ($N = 1024$), there is no *border effect* and the MSE is almost the same for all the sub-carriers. However, when null sub-carriers are inserted, the MSE of the sub-carriers at the edge of the spectrum is degraded leading to the *border effect* phenomenon. We notice also that the impact of *border effect* increases with the number of null sub-carriers. Using TSVD method, we can see that the *border effect* phenomenon is significantly reduced and the MSE at the edge of the spectrum is improved. This is due to the reduction of the condition number provided by TSVD. Therefore, TSVD-based channel estimation is considered in the rest of the section.

Table 5.2: Characteristics of LTE channel models.

	EPA	EVA		ETU	
Number of paths	7	9		9	
τ_{max} [ns]	410	2510		5000	
τ_{rms} [ns]	43	357		991	
B_c[KHz] (1.11)	23256 (1550 sc)	2800 (186 sc)		1009 (67 sc)	
$B_{c50\%}$[KHz](1.12)	4651 (310 sc)	560 (37 sc)		202 (13 sc)	
$B_{c90\%}$[KHz] (1.12)	465 (31 sc)	56 (3 sc)		20 (< 2 sc)	
f_m [Hz]	5	5	70	70	300
v[Km/h]	2	2	30	30	130
T_{c1} [ms](1.15)	200 (2799 symb)	200 (2799 symb)	14.3 (200 symb)	14.3 (200 symb)	3.3 (48 symb)
T_{c2} [ms] (1.16)	36 (503 symb)	36 (503 symb)	2.6 (36 symb)	2.6 (36 symb)	0.6(8 symb)
T_{c3} [ms] (1.16)	85 (1190 symb)	85 (1190 symb)	6 (84 symb)	6 (84 symb)	1.4 (19 symb)

Table 5.3: Channel estimation techniques.

Perfect	Perfect knowledge of the channel
LS	LS estimation at the pilot symbols followed by a linear interpolation
LMMSE	Winner filtering in time and frequency domain
DFT	Classical DFT channel estimation
TSVD	DFT channel estimation using TSVD

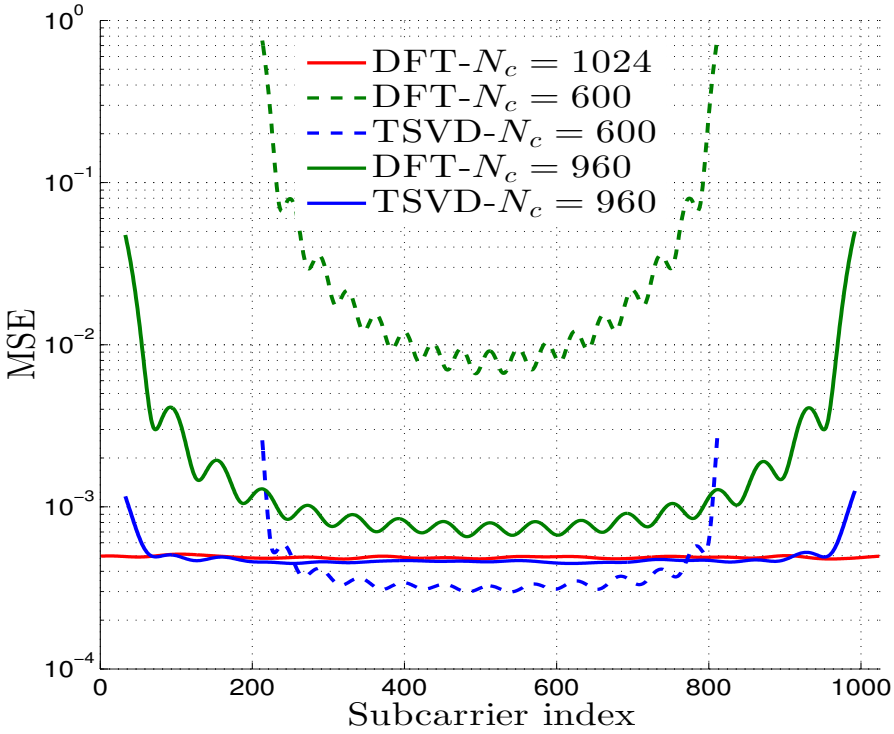


Figure 5.8: MSE in function of sub-carriers index for classical DFT and TSVD based channel estimation with $N = 1024$, $N_c = 1024$, 960, 600 at SNR = 20 dB, 4-QAM.

Figure 5.9 shows BER performance of channel estimation methods in a 2×2 spatially multiplexing system with 4-QAM and 16-QAM in EPA channel model. We can see that LMMSE shows a performance degradation of 0.25 dB at a BER level of 1×10^{-4} compared to perfect channel knowledge. Meanwhile, a performance degradation of 2 dB is observed with LS estimates. Moreover, TSVD presents a performance degradation of 0.25 dB compared to LMMSE and 0.5 dB compared to the perfect case in the case of 4-QAM. However, in the case of 16-QAM, TSVD presents approximately similar performance as LMMSE with a degradation of 0.5 dB compared to perfect channel.

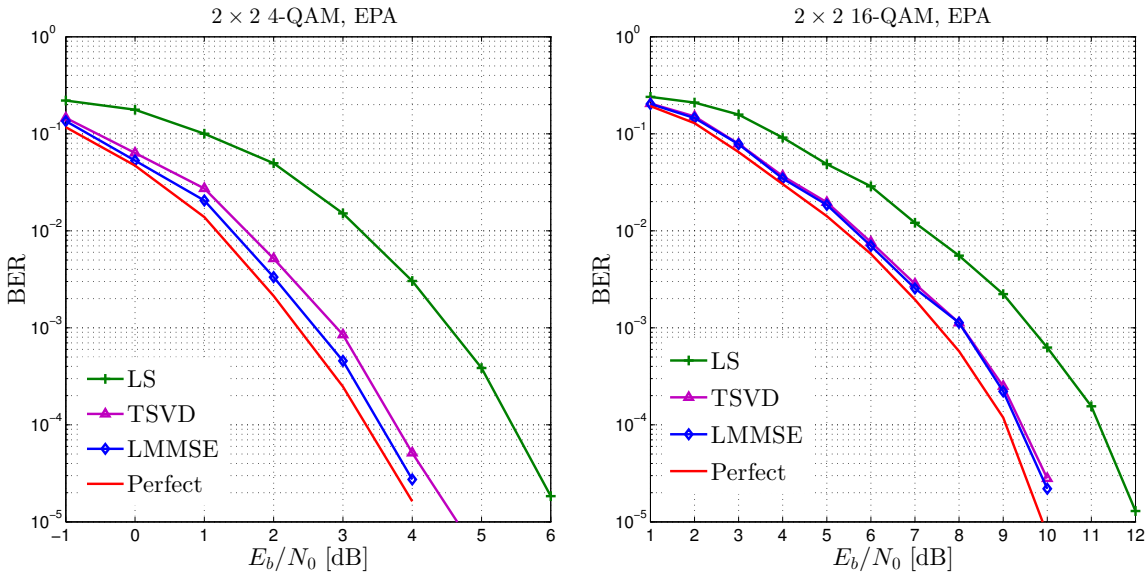


Figure 5.9: BER performance of a $\mathbf{2 \times 2}$ SM system using several CE methods (LS, LMMSE, TSVD) on EPA channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

With the increase of the number of antennas to 4, less pilots are inserted for the 3rd and 4th antennas as illustrated in Figure 5.3 in order to reduce the loss of spectral efficiency. Figure 5.11 depicts the performance of channel estimation methods in the case of 4×4 system with 4-QAM and 16-QAM in the EPA channel. We can see that channel estimation methods present performance degradation compared to the case of 2×2 system. LMMSE presents a performance degradation of 0.5 dB compared to the perfect channel. However LS method shows a significant performance degradation more than 4 dB at a BER level of 1×10^{-4}. The performance degradation of TSVD is 0.5 dB compared to LMMSE and 1 dB compared to perfect channel knowledge.

Figure 5.10 shows the corresponding MSE. It is clear that LMMSE presents a lowest MSE following by TSVD and LS.

EPA channel model can be considered as a flat fading and slow varying environment as shown in Table 5.2. In this case, TSVD seems to be attractive since it greatly improves the accuracy of LS estimates and approaches the performance of LMMSE with significant reduction in the computational complexity.

Let us now study the performance of channel estimation methods in more frequency and time selective channels. Figure 5.12 shows the performance of CE in EVA channel

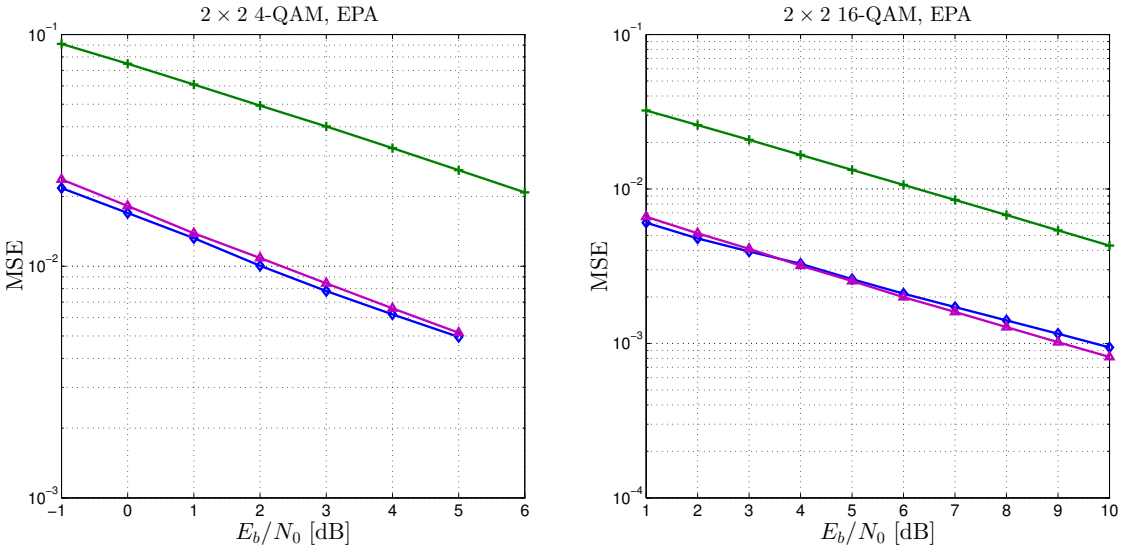


Figure 5.10: MSE versus E_b/N_0 of a $\mathbf{2 \times 2}$ SM system using several CE methods (LS, LMMSE, TSVD) on EPA channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

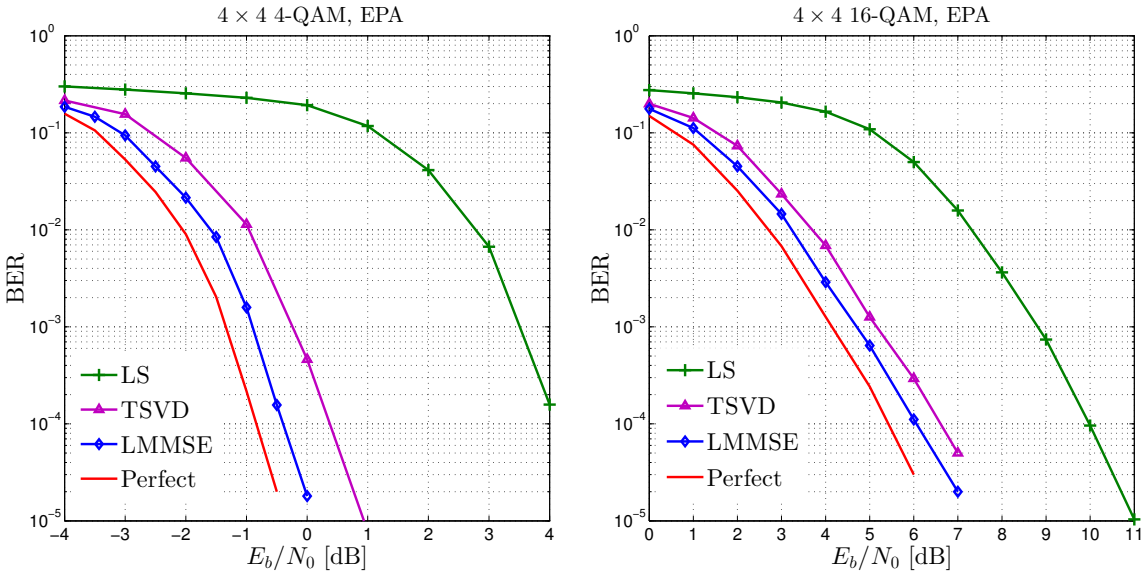


Figure 5.11: BER performance of a $\mathbf{4 \times 4}$ SM system using several CE methods (LS, LMMSE, TSVD) on EPA channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

model in the case of 2×2 system with 4-QAM and 16-QAM. Two Doppler frequencies ($f_m = 5$ Hz and $f_m = 70$ Hz) are considered. These frequencies correspond to a low speed user velocity ($v = 2$ km/h) and medium speed user velocity ($v = 30$ km/h), respectively. Solid curves correspond to the slow fading case ($f_m = 5$ Hz), and dashed curves correspond to the medium speed case ($f_m = 70$ Hz). With $f_m = 5$ Hz, similar behavior can be observed as in EPA channel. A performance degradation of 0.5 dB, and 2 dB is observed with LMMSE and LS respectively. Meanwhile, TSVD presents a performance degradation of 1 dB and 0.75 dB with 4-QAM and 16-QAM, respectively.

However, with $f_m = 70$ Hz, more performance loss is observed due to the variability of the channel. In the case of 4-QAM, the performance degradation is about 1 dB, 1.2 dB and 2.5 dB with LMMSE, TSVD, LS, respectively. However in the case of 16-QAM, we can see that the performance loss is more significant. LMMSE and TSVD present a degradation of 1.6 dB, while the loss is 3 dB with LS estimates.

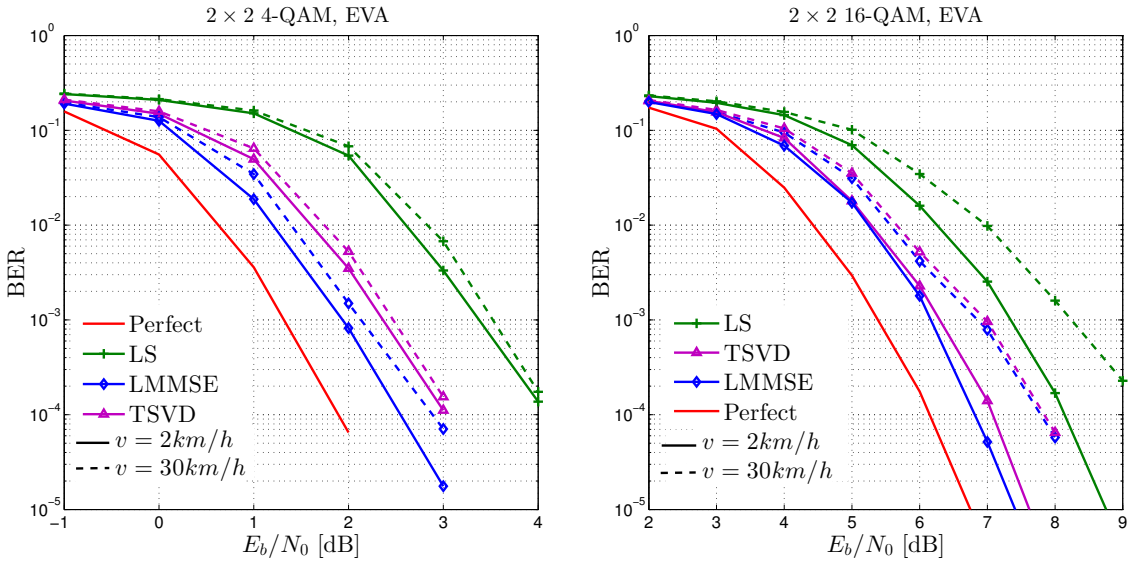


Figure 5.12: BER performance of a $\mathbf{2 \times 2}$ SM system using several CE methods (LS, LMMSE, TSVD) on EVA channel, with **(a)** 4-QAM, and **(b)** 16-QAM. $N = 1024$, $N_c = 600$.

In the case of 4×4 system, the performance degradation becomes more severe as illustrated in Figure 5.13. For the low speed case $f_m = 5$ Hz, LMMSE presents a performance degradation of 1.5 dB and 2 dB with 4-QAM and 16-QAM, respectively. The performance loss is about 2.5 dB and more than 4 dB with TSVD and LS estimates receptively. For $f_m = 70$ Hz with 4-QAM, the performance degradation increases to 2 dB, 3 dB and more than 4 dB with LMMSE, TSVD and LS estimates, respectively. Meanwhile, this degradation significantly increases in the case of 16-QAM to more than 4 dB with LMMSE and TSVD and more than 8 dB with LS method.

Additionally, we consider a more time varying channel. Figure 5.14 depicts the performance of 2×2 system with 4-QAM and 16-QAM on ETU channel. In the case of $f_m = 70$ Hz with 4-QAM, we show that the performance degradation is about 1.5 dB, 1.6 dB and 2.2 dB with LMMSE, TSVD and LS estimates respectively. However, with 16-QAM, the performance degradation becomes more significant. LMMSE and TSVD present a degradation of 2.75 dB and 3 dB respectively, with more than 5 dB of loss with LS. In the case of high Doppler frequency, the performance degradation increases with 4-QAM to 2 dB, 2.5 dB and 4 dB with LMMSE, TSVD and LS, respectively. Meanwhile in the case of 16-QAM, the situation becomes more severe and a performance loss of 5 dB and more is observed with different CE methods.

Similarly, a 4×4 system is considered in Figure 5.15. Same remarks can be retrieved from this figure compared to 2×2 with more performance loss. We can also see that

in case of 16-QAM, an error floor is obtained with various CE methods (more than 7 dB at a BER level of 2×10^{-2}).

In order to summarize the performance of channel estimation methods on different channel environments, we provide the performance loss of these CE methods compared to perfect channel knowledge at a BER level of 1×10^{-4} in Table 5.4.

Through our simulation results, we can see that TSVD is a well suitable solution for channel estimation in a slow fading channels (indoor environment) since it is able to significantly reduce the noise effect and improve the performance compared to LS. In addition, it provides near LMMSE channel estimate (0.5 dB of performance degradation) with a reduced complexity. Meanwhile, in more selective scenarios, pilot-based channel estimation methods are not able to track the variation of the channels and suffers from a significant performance degradation. Consequently, in order to achieve better channel estimation accuracy, more advanced CE methods must be used (DD, CS) at a cost of an increase of the computational complexity. For example, the channel estimation methods can be incorporated into the iterative detection and decoding in order to improve the accuracy of channel coefficients using the estimated symbols.

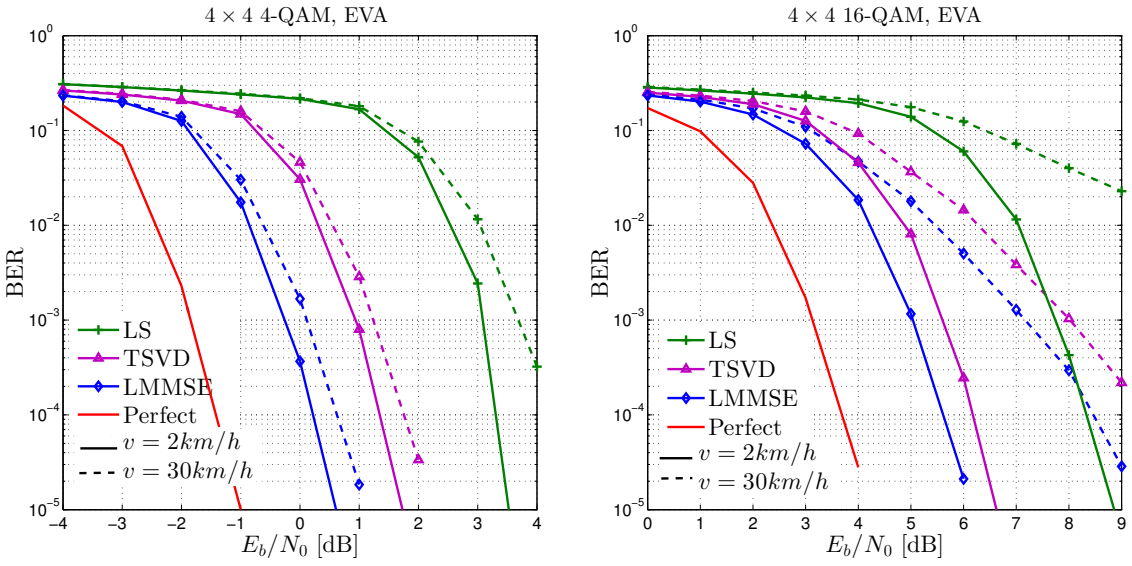


Figure 5.13: BER performance of a $\mathbf{4 \times 4}$ SM system using several CE methods (LS, LMMSE, TSVD) on EVA channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

5.8 Testbed for MIMO-OFDM system in real-time environment

This section describes a testbed for the MIMO-OFDM system using WARP platform for the purpose of validating the performance of the iterative receiver in real time environments. The platform allows studying the impact of RF module on the performance of the system as well as synchronization and channel estimation effect. First, the synchronization for MIMO-OFDM system is presented. We then describe the characteristics of WARP platform [208]. Consequently, the system setup used to validate

the performance of the iterative receiver is introduced. At the end, different scenarios and antennas configurations are considered in addition to some useful measurements.

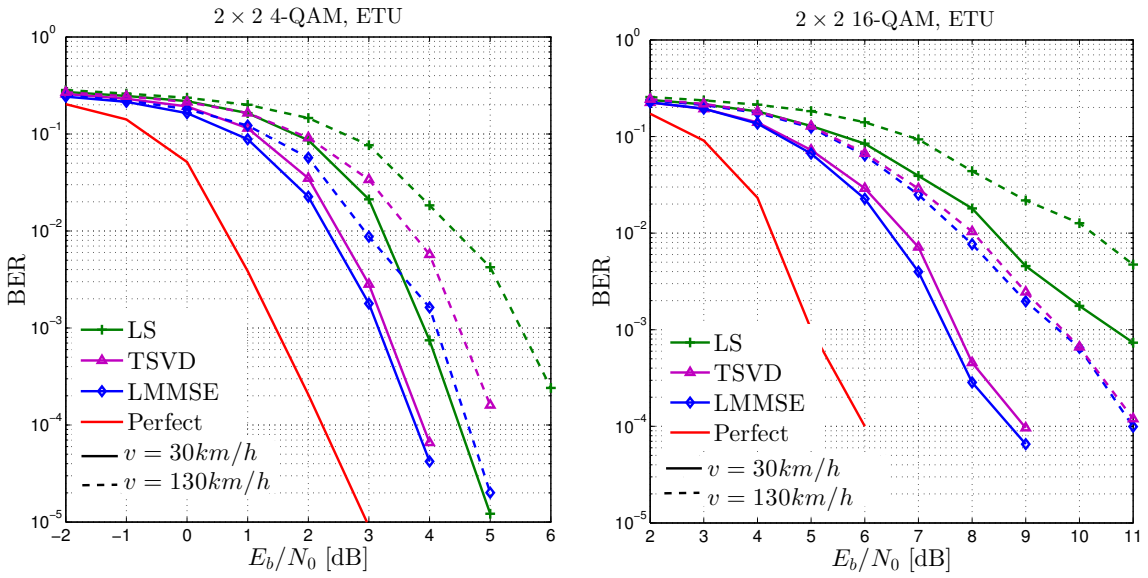


Figure 5.14: BER performance of a $\mathbf{2 \times 2}$ SM system using several CE methods (LS, LMMSE, TSVD) on ETU channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

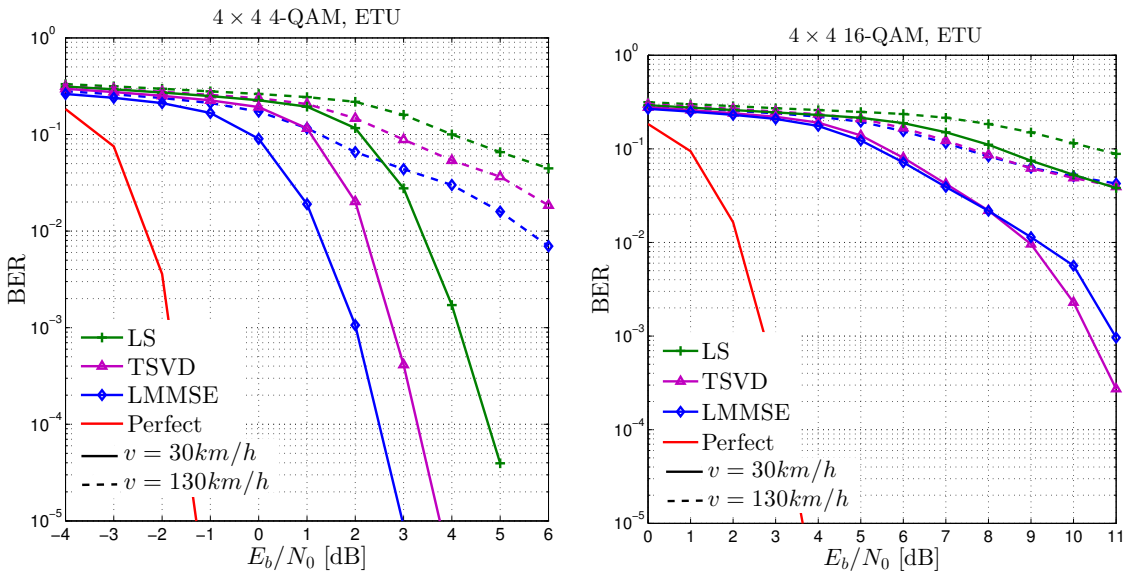


Figure 5.15: BER performance of a $\mathbf{4 \times 4}$ SM system using several CE methods (LS, LMMSE, TSVD) on ETU channel, with **(a)** 4-QAM, and **(b)** 16-QAM, $N = 1024$, $N_c = 600$.

5.8.1 Synchronization in MIMO-OFDM system for the testbed

In MIMO-OFDM system, the synchronization is a critical issue since different transmitters need to be synchronized to different receivers. A small synchronization error

Table 5.4: Performance loss of various CE methods compared to perfect channel at a BER level of 1×10^{-4} in 2×2 and 4×4 SM system with 4-QAM and 16-QAM.

	2×2 4-QAM			2×2 16-QAM		
	LS	LMMSE	TSVD	LS	LMMSE	TSVD
EPA	2 dB	0.25 dB	0.5 dB	2 dB	0.25 dB	0.3 dB
EVA ($f_m = 5$ Hz)	≈ 2 dB	≈ 0.5 dB	1 dB	2 dB	0.5 dB	0.75 dB
EVA ($f_m = 70$ Hz)	> 2 dB	1 dB	1.2 dB	≈ 3 dB	1.6 dB	1.6 dB
ETU ($f_m = 70$ Hz)	2.5 dB	1.5 dB	1.6 dB	> 5 dB	2.75 dB	3 dB
ETU ($f_m = 300$ Hz)	> 4 dB	2 dB	2.5 dB	> 8 dB	5 dB	5dB
	4×4 4-QAM			4×4 16-QAM		
	LS	LMMSE	TSVD	LS	LMMSE	TSVD
EPA	> 4 dB	0.5 dB	1.2 dB	> 4 dB	0.5 dB	1 dB
EVA ($f_m = 5$ Hz)	> 4 dB	1.5 dB	2.5 dB	> 4 dB	2 dB	2.5 dB
EVA ($f_m = 70$ Hz)	> 5 dB	2 dB	3 dB	> 8 dB	> 4 dB	> 5 dB
ETU ($f_m = 70$ Hz)	6 dB	4 dB	4.7 dB	> 8 dB	> 8 dB	> 8 dB
ETU ($f_m = 300$ Hz)	> 8 dB	> 8 dB	> 8 dB	> 8 dB	> 8 dB	> 8 dB

would lead to the loss of orthogonality between sub-carriers and therefore degrade the system performance. The conventional synchronization must be performed in both time and frequency domains [209, 41, 210, 211].

The time synchronization is done in two steps, frame detection or coarse synchronization and fine synchronization. The coarse synchronization consists in finding the start of the frame over an approximate range of samples. Meanwhile, the fine synchronization consists in accurately finding the start of the useful portion of the frame. The frequency synchronization consists of frequency offset estimation followed by a fine frequency offset correction. Normally, the carrier frequency generated by the local oscillator at the receiver is not exactly the same as the carrier frequency generated by the local oscillator at the transmitter. This creates a carrier frequency offset (CFO) in the baseband signal. The frequency synchronization is very important for OFDM systems because a small frequency offset would cause loss of orthogonality between the sub-carriers and hence causing Inter Carrier Interference (ICI).

In general, the synchronization is sequentially accomplished by coarse time synchronization, frequency offset estimation, and fine frequency offset estimation and fine time synchronization. In our testbed (cf. section 5.8), same local oscillator is used at the transmitter and at the receiver, the frequency offset is zero. In the rest of this section, time synchronization will be discussed.

5.8.1.1 Time synchronization

The main task of the time synchronization is to detect the frame arrival instant in the received signals. The start of the frame can be detected based on the power level of the receive signals. If the power level is higher than a given threshold, then we can assume the start of the frame. However, this method is inefficient in noisy channels, since noise

power will be amplified at the reception by the automatic gain control (AGC) circuit. Usually, the time synchronization is performed using some special kind of training sequences. The synchronization sequence can be generated either in time domain or in frequency domain. In our case, the preamble is generated in the frequency domain. Meanwhile, at the receiver, the synchronization is performed in the time domain.

The time synchronization is generally achieved in two steps, namely coarse synchronization and fine synchronization. Coarse time synchronization can be performed by auto-correlating the received samples over a window of CP samples that are at a distant of N_{seq} from each other, where N_{seq} is the period of the synchronization sequence assumed to be equal to the FFT size ($N = N_{\text{seq}}$). Another method consists in repeating M times the training sequence so that $N = MN_{\text{seq}}$. In this case, a running correlation window of every N_{seq} samples with the next consecutive N_{seq} samples can be performed to get the auto-correlation peak corresponding to the beginning of the sequence. The fine synchronization is performed after the frequency offset correction by cross-correlating the received samples with a well-known training sequence.

Several kinds of training sequences have been proposed in the literature in both the frequency and time domains. Same examples include pseudo noise (PN) sequences, constant amplitude zero auto-correlation (CAZAC) sequences, Barker sequences, Hadamard sequences and Gold sequences. The sequences should have very good correlation properties. Herein, CAZAC sequences are considered and especially Zadoff-Chu sequences [212]. CAZAC sequence of length N_{seq} is given by:

$$C_u(k) = \begin{cases} \exp \frac{j\pi uk^2}{N_{\text{seq}}} & \text{if } N_{\text{seq}} \text{ is even,} \\ \exp \frac{j\pi uk(k+1)}{N_{\text{seq}}} & \text{if } N_{\text{seq}} \text{ is odd,} \end{cases} \tag{5.26}$$

where $k = 0, ..., N_{\text{seq}}$, and u is the root and must be prime factor of N_{seq}. CAZAC sequences are a class of complex-valued sequences with zero cyclic auto-correlation. It means that CAZAC code is always orthogonal with its cyclic shifted versions.

In the case of MIMO-OFDM system, the receive signal is a combination of the signals transmitted from all transmit antennas. Therefore, orthogonal sequences must be used, for example cyclically shifted CAZAC sequences can be used to achieve synchronization. Because of the good auto-correlation property, these cyclically shifted sequences would give a cross-correlation output of zeros except at the multiples of the cyclic shift distance.

5.8.2 Platform description

Wireless Open-Access Research Platform (WARP) is a scalable and extensible programmable platform built to prototype advanced wireless networks [208]. It allows the researcher to flexibly modify the transmission procedure on all layers, ranging from the physical layer up to the application layer. WARP v3 platform is the latest generation of WARP hardware. It consists of a Xilinx Virtex-6 FPGA, two flexible RF interfaces and multiple peripherals to facilitate rapid prototyping of custom wireless designs as depicted in Figure 5.16. The corresponding block diagram of the hardware design is illustrated in Figure 5.17.

Virtex-6 FPGA includes a large number of built-in system-level blocks. It allows high-performance logic designs, high-performance DSP designs, and high-performance

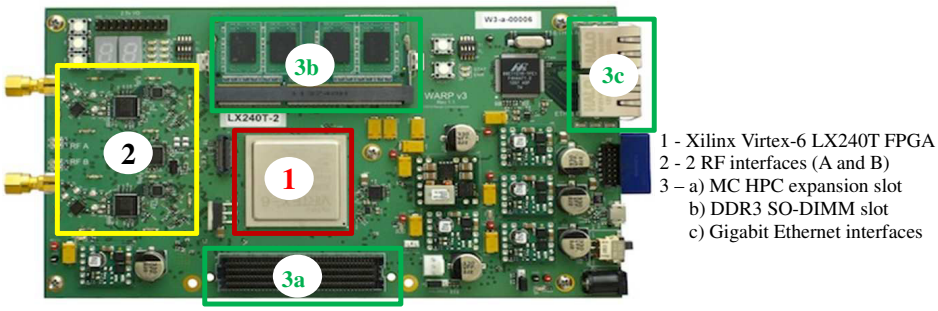


Figure 5.16: WARP v3 hardware.

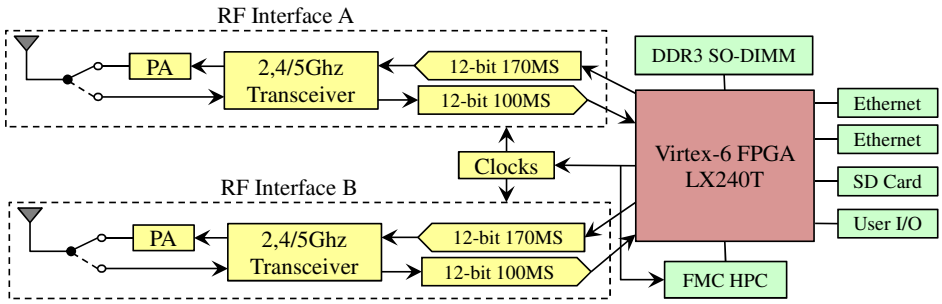


Figure 5.17: Block diagram of the WARP v3 hardware.

embedded systems designs with unprecedented logic, DSP, connectivity, and soft microprocessor capabilities [213]. Three FPGA configuration methods are provided by WARP v3 board namely JTAG, SD card and SPI flash.
The RF interfaces consist of a dual-band power amplifier (20 dBm Tx power), an 2.4/5 GHz transceiver with a bandwidth up to 40 MHz, an 12-bits digital to analog converter (DAC) with a sampling rate up to 170Msps and an 12-bits analog to digital converter (ADC) with a sampling rate up to 100 Msps.
DDR3 SO-DIMM memory interface is designed to support modules up to 8 GB in size and run at up to 400 MHz. The gigabit Ethernet serves as the interface between the board and the wired internet. The WARP v3 has also an a high-pin count (HPC) FMC expansion slot to allow multiple banks of signals to be connected to the FPGA on single board. In our work, an FMC-RF-2X245b module with two dual-band RF interfaces is used to provide up to 4 antennas on a single board. Variety of user I/O is included in the board in order to observe and interact with designs at run time. The user I/O includes 12 LEDs, 3 push buttons, 4-position DIP switches and 2 seven-segment displays.

5.8.3 System imperfections

For the system operating in real-time conditions, several factors have an impact on the performance of the system. Therefore, these factors must be taken into account in the design of the system. Among these factors, we have:

- DAC/ADC effect: The quantization of the data into a fixed-point format (12 bits) causes truncation of the signals and loss of the accuracy.
- Non linearity of the power amplifier for signals with high PAPR as OFDM signal.
- Channel estimation error: The channel accuracy depends on channel estimation

methods and interpolation methods.
- Time synchronization: to detect the start of the frame.
- Frequency synchronization: to correct the frequency offset between local oscillators at the transmitter and the receiver.
- Symbol time recovery: If the sampling rate at the reception is not recovered correctly, it may lead to the loss of some samples. This can be avoided by using the same clock between the transmitter and the receiver.

We note that the impact of these factors on the performance of the system is not considered during simulations. However, such a real-time testbed allows us to study the effect of the RF modules (ADC, DAC, amplifier, filter, etc.), channel estimation and synchronization.

5.8.4 System setup

In our system setup, a 2×2 MIMO-OFDM system is considered as depicted in Figure 5.18. The block diagram of the transmitter and the receiver including the RF components is illustrated in Figure 5.19. Table 5.5 summarizes the system setup parameters.

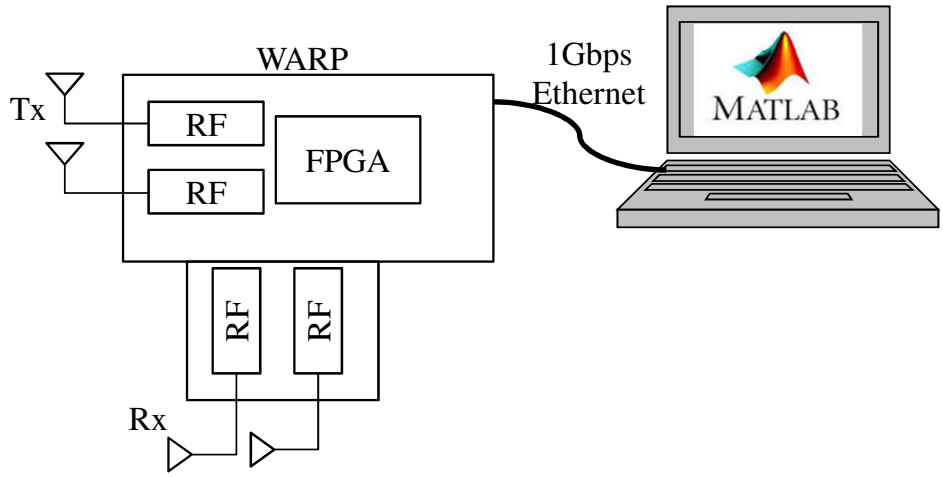


Figure 5.18: System setup with $N_t = N_r = 2$.

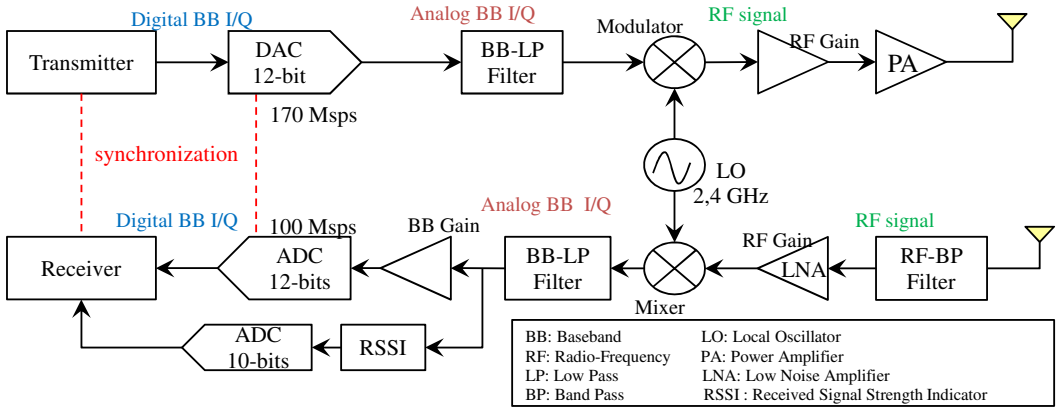


Figure 5.19: Block diagram of the transmitter and the receiver with the RF components.

Table 5.5: System setup Parameters.

MIMO system	2×2 SM
Number of sub-carriers $N(N_c)$	1024 (600 used) samples
Cyclic Prefix (CP)	80 samples
Number of OFDM symbol	$N_{symb} = 12$
Sampling frequency f_e	40 MHz
Carrier frequency f_c	2.484 GHz
Preamble duration (samples)	$N_t(CP + N) = 2(1024 + 80)$

At the transmitter side, the data symbols and the training symbols (preamble) are generated by a host computer using Matlab, which results into N_t frames of baseband I/Q signals. The frame structure with $N_t = 2$ is represented in Figure 5.20. The preamble consists of $N_t = 2$ training symbols of length $N + CP$ designed for time synchronization and channel estimation. The synchronization sequence is generated from a frequency domain Zadoff-Chu sequence. The data consists of $N_{symb} = 12$ OFDM symbols. Dummy bits are added at the end of the frame to fill the buffer of 2^{14} samples of the WARP board.

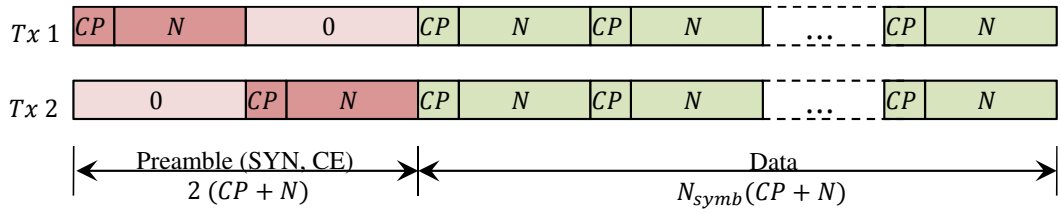


Figure 5.20: Structure of the transmitted frame.

The digital baseband I/Q samples are converted into analog baseband I/Q signals by the DACs. The carrier frequency $f_c = 2.484$ GHz is generated using the LO. Then the analog I/Q signals are up-converted into RF signals. The output signal is amplified using a power amplifier (PA). Subsequently, the signal is transmitted in burst mode or in continuous mode over real-time channel environment. The channel is assumed to be constant over the transmission of the frame.

At the receiver side, a superposition of the transmitted signal is received. We assume that the signals from each transmit antennas are reached at the same time by the receiver. The analog RF received signals are then filtered. The filtered RF signal is down-converted into the analog baseband I/Q signals. AGC is used to adjust the gain of the receive antennas. The AGC controls the signal power in order to avoid the overflows at ADC and to optimally exploit the dynamic range of the digital baseband signals. Then, the digital signal is passed to a host computer for processing using Matlab.

We note that DAC and ADC must be synchronized. If ADC and DAC have different sampling clocks, then the mismatch in the clock speed would lead to either additional samples or loss of samples depending upon the relative speed between the two clocks.

Additionally, frequency synchronization, time synchronization and channel estimation must be performed before the detection of the transmitted signals. The synchronization and the channel estimation processes are closely related issues. Once the fine synchronization is achieved, the synchronization pilots can be used for channel estimation. However, frequency synchronization is required when there is a mismatch between the transmitter and receiver local oscillators. But in our case, we use the same local oscillator in the transmitter and the receiver sides, the frequency offset is zero. Due to the absence of frequency offset, the time synchronization can be simply performed at each antenna by cross-correlating its received sequence with the known training sequence as illustrated in Figure 5.21. The channel estimation is then performed using least square (LS) principle described in section (5.4.1). To avoid interference between antennas, a simplest way is used. It consists in transmitting null symbols on one antenna and the training symbols on the other one as illustrated in Figure 5.20.

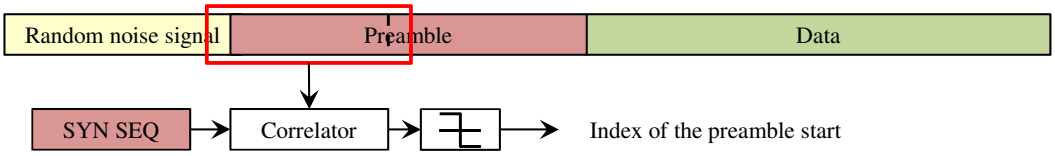


Figure 5.21: Cross-correlation of the received signal and the synchronization sequence.

5.8.5 Testbed results

In this section, we carry out simulations and test measurements to demonstrate the performance of the proposed iterative receiver in real-time conditions with different constellations and channel coding schemes.

Several antenna configurations are considered as listed in Table 5.6. The first configuration is a parallel configuration where the antennas are placed at an equal distance of 35 cm with a same polarization. The second configuration is orthogonal, where the transmit antennas and receive antennas are positioned perpendicularly at an equal distance of 35 cm. The last configuration is an arbitrary one, where the antennas are more spaced (distance of 1.5 m).

Table 5.6: Antenna configurations.

Config. 1	Parallel antenna configuration, distance = 35 cm
Config. 2	Orthogonal antenna configuration, distance = 35 cm
Config. 3	Arbitray antenna configuration, distance = 1.5 m

The transmitted signals are captured by an oscilloscope as shown in Figure 5.22. The frame duration T_{frame} without the dummy bits is equal to 386.4 μs. This duration corresponds to the product of the number of samples $14 * (N + CP)$ and the sampling period ($f_e = 40$ MHz, $T_e = 25$ ns). The spectrum occupied by the transmitted signals is represented in Figure 5.23. The occupied bandwidth is about 23.5 MHz when only data symbols are transmitted. However 40 MHz bandwidth is occupied when the preamble is transmitted. This is explained by the fact that only $N_c = 600$ data sub-carriers are transmitted, while in the preamble, all the sub-carriers $N = 1024$ are used for the

synchronization sequence. We can also see that there are also some pics even with no transmission which correspond to other wireless transmission, since this band is an industrial, scientific and medical (ISM) radio band.

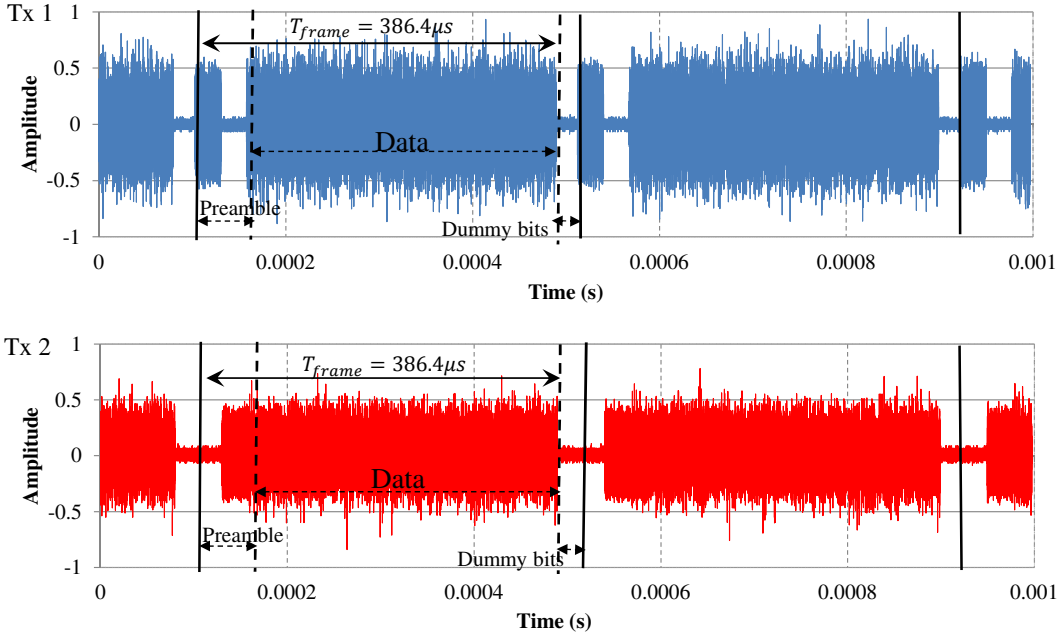


Figure 5.22: Transmitted signals.

Table 5.7 shows the estimated SNR and the start of the preamble of the received frame in the case of turbo decoder and LDPC decoder with different constellations. The obtained BER is zero in all these cases. The start of the preamble is 38 samples when the antennas are placed at a distance of 35 cm. When the antennas are more spaced (distance of 1.5 m), this index increases to 39 samples since the time delay increases. Moreover, the estimated SNR depends also on the distance between the antennas. The SNR is in the range of [22-25] dB in the first two cases. In the last case, the estimated SNR is about [30-33] dB. This is due to the increase of the antenna gain that increases the power of the received signal.

Table 5.7: Estimated SNR and start of the preamble of the received frame with turbo decoder and LDPC decoder for **2 × 2** MIMO system with 4-QAM, 16-QAM and 64-QAM.

	Config. 1			Config. 2			Config. 3		
	Turbo decode								
	4-QAM	16-QAM	64QAM	4-QAM	16-QAM	64QAM	4-QAM	16-QAM	64QAM
SNR [dB]	23.28	23.41	25.32	25.21	24.91	26.88	30.56	31.94	32
Start Preamble	38	38	38	38	38	38	39	39	39
	LDPC decoder								
	4-QAM	16-QAM	64QAM	4-QAM	16-QAM	64QAM	4-QAM	16-QAM	64QAM
SNR [dB]	22	22.40	22.89	24.60	24.91	26.25	29.88	32.78	32.28
Start Preamble	38	38	38	38	38	38	39	39	39

Figures 5.24 show examples of the received constellation and the detected constellation in the case of 4-QAM, 16-QAM and 64-QAM. In Figures 5.25 and 5.26, examples of

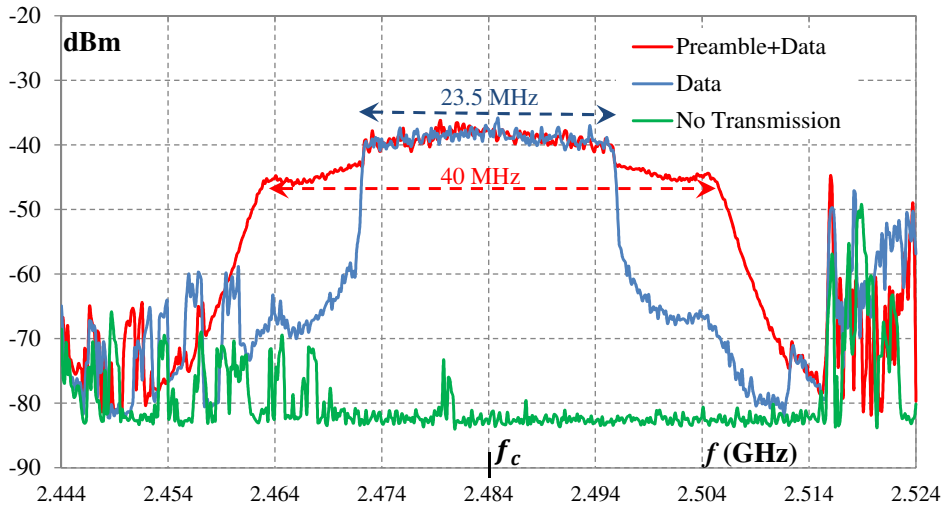


Figure 5.23: Frequency spectrum.

the channel frequency response and the channel impulse response between the transmit and receive antennas are illustrated. The indoor channel can be modeled by 5 significant paths.

We can see that even with high-order constellation, the iterative receiver is able to recover the transmitted signal from interference. In these scenarios, the AGC is used to automatically adapt the gain of the received antennas, which gives a good SNR. In order to test the performance of the receiver in low SNR regime, we manually set the gain of the received antennas. Then, we change the number of inner and outer iterations of LC-K-Best based receiver to view their impact on BER performance. Table 5.8 lists the obtained BER and the estimated SNR with different number of inner and outer iterations in the case of 4-QAM and 16-QAM with turbo decoder. The number in the parenthesis (I_{out}, I_{in}) corresponds to the number of outer and inner iterations, respectively. We can see that (4,2) configuration achieves the lowest BER. This results validate the results in the previous chapters and enable us to test the performance of the proposed iterative receiver in real time environments.

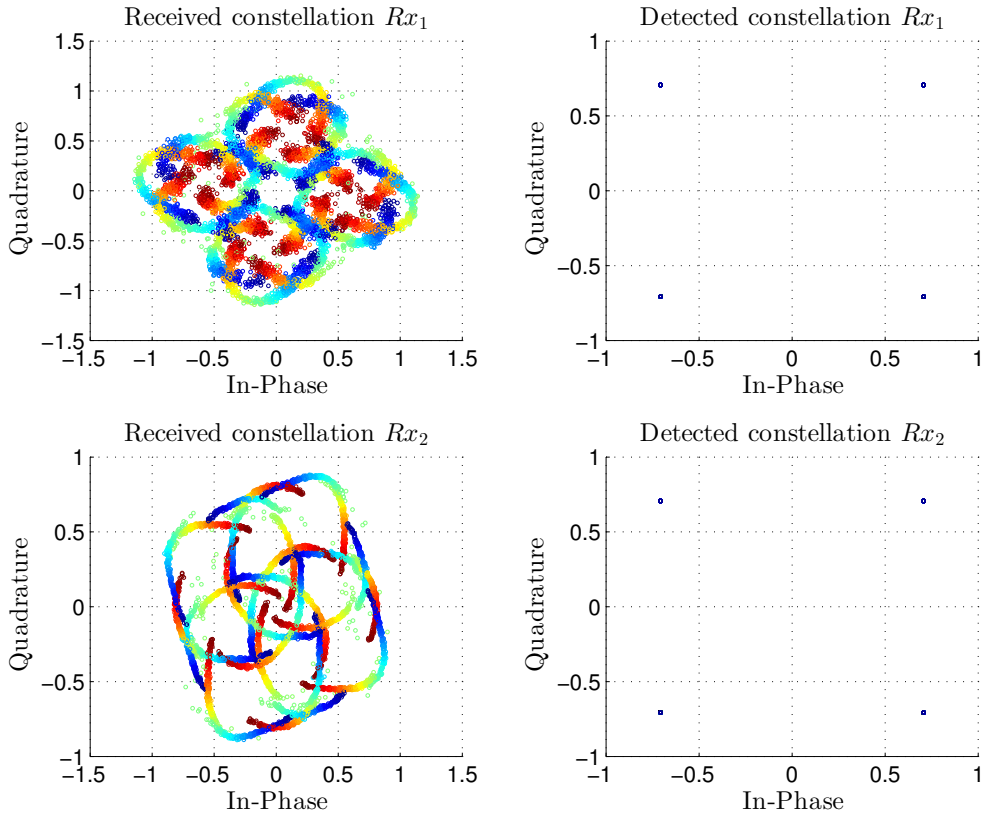
Table 5.8: BER of the iterative receiver with turbo decoder for $\mathbf{2 \times 2}$ MIMO system with 4-QAM, 16-QAM, (I_{out}, I_{in}).

			4-QAM		
	(4,2)	(3,2)	(2,2)	(1,2)	(1,8)
BER	5.8×10^{-4}	4.2×10^{-3}	4×10^{-3}	1.5×10^{-2}	1.4×10^{-2}
SNR [dB]	9.71	9.13	9.21	9.0	9.0
			16-QAM		
	(4,2)	(3,2)	(2,2)	(1,2)	(1,8)
BER	0.01	0.029	0.06	0.027	0.021
SNR [dB]	13.41	13.14	12.65	13.15	13.48

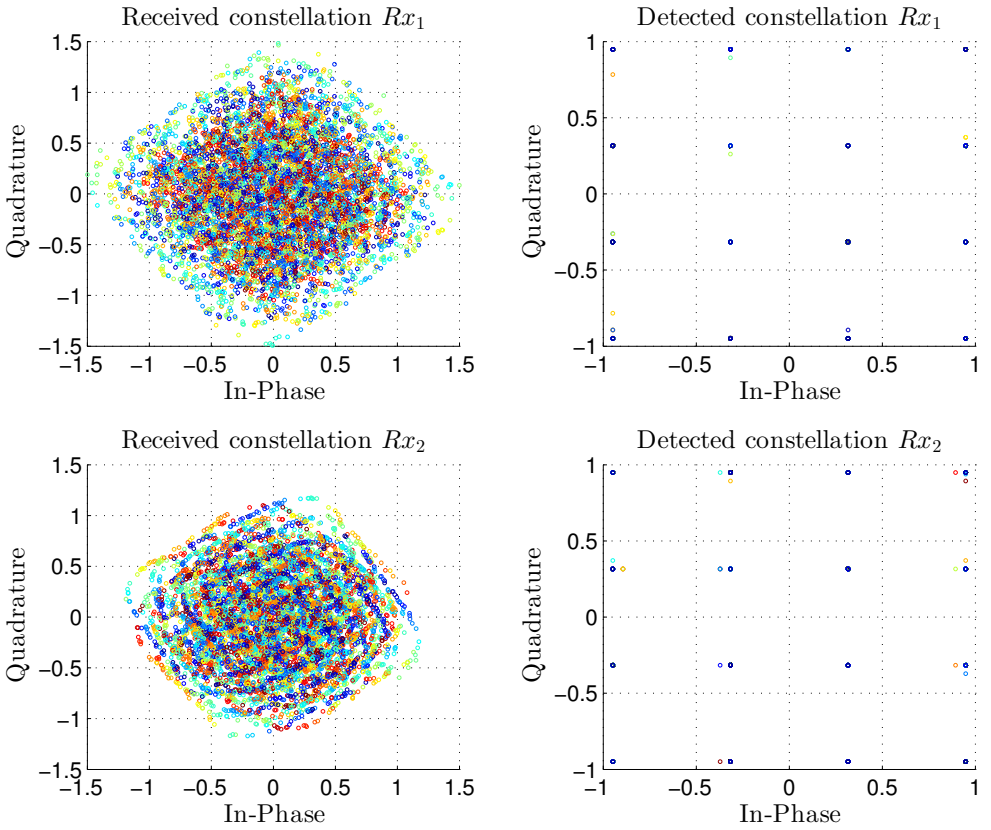
5.9 Conclusion

In this chapter, we have presented some issues encountered by the receiver in practical realization of MIMO-OFDM system. First, several channel estimation techniques for MIMO-OFDM system have been investigated. We have described the commonly used pilot-based methods namely LS and LMMSE. Then the transform domain methods namely DFT and TSVD have been presented. Their performances have been compared in several LTE channel models. Additionally, the synchronization of MIMO-OFDM system has been briefly reviewed. At the end, a testbed using WARP platform has been used to validate the performance of the proposed receiver in real-time.

(a) 4-QAM



(b) 16-QAM



(c) 64-QAM

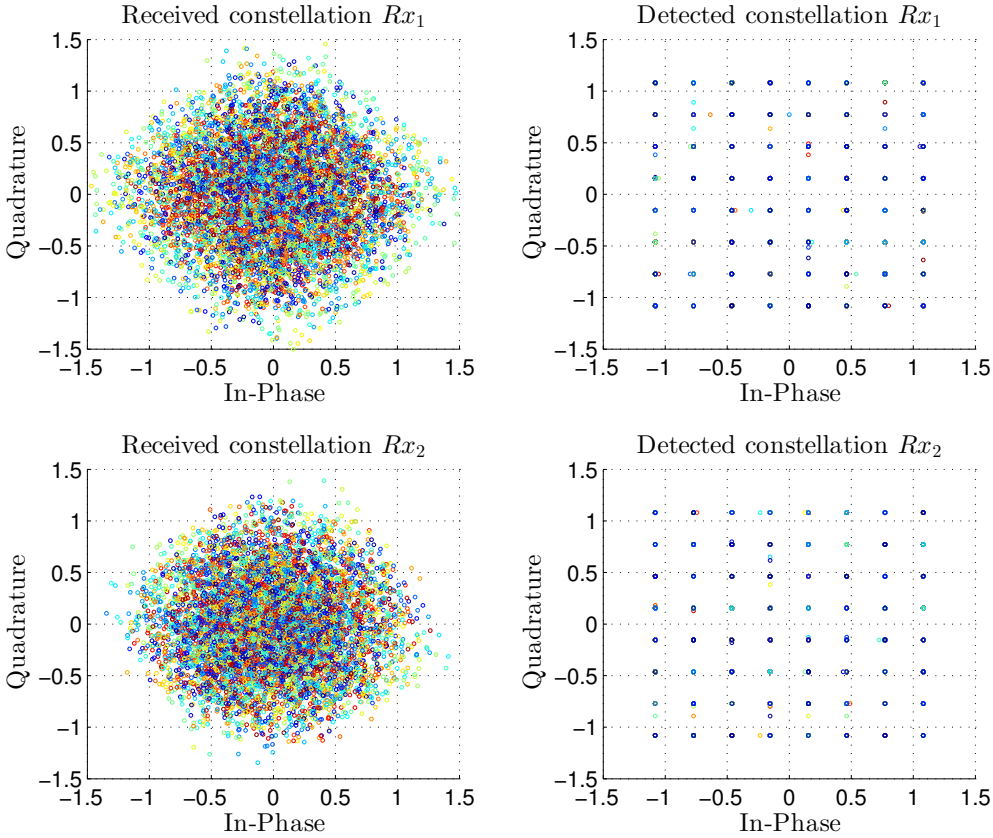


Figure 5.24: Received and detection constellations in the case of (a) 4-QAM, (b) 16-QAM and (c) 64-QAM.

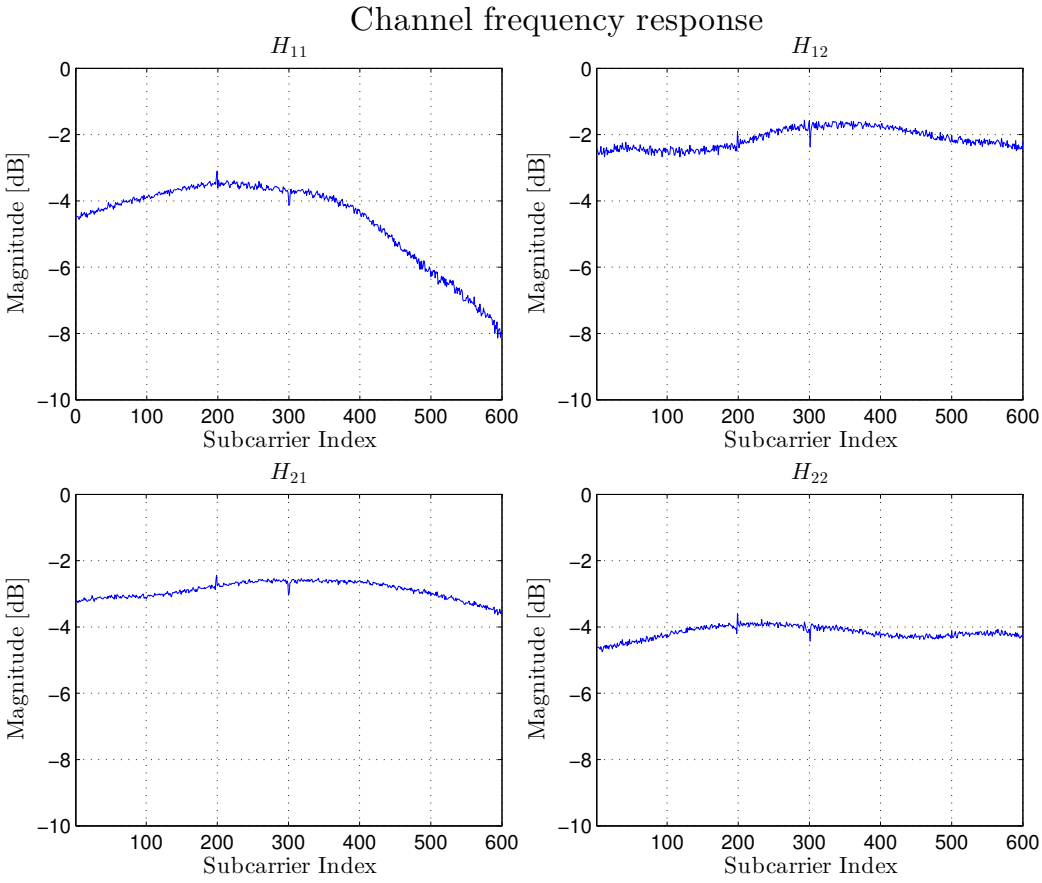


Figure 5.25: Channel frequency response for 2×2 MIMO system.

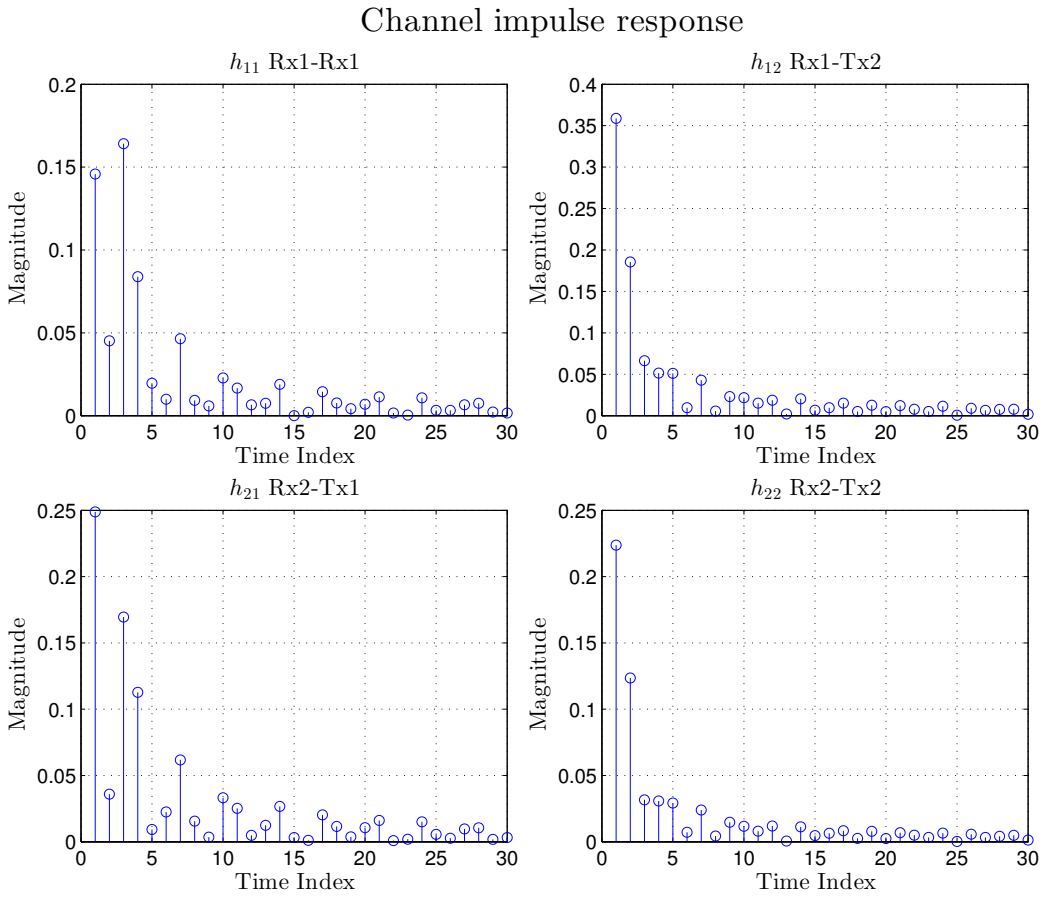


Figure 5.26: Channel impulse response for 2×2 MIMO system.

Conclusions & Future works

In recent years, iterative receiver was widely considered since it is able to approach the performance limits in emerging wireless communication systems. However, such a receiver is extremely challenging in practical implementation due to the complexity associated with MIMO detection and channel decoding.

In this thesis, the convergence, the performance and the complexity of iterative receiver combining MIMO detection with channel decoding have been investigated. The aim was to develop algorithms and architectures to meet the high performance and low computational complexity requirements of future wireless communication systems.

Firstly, the fundamental principles of wireless communication system as well as the channel characteristics were summarized in chapter 1. The channel coding schemes including turbo code and LDPC code, OFDM and MIMO techniques were also considered. The system model and the detection problematic have been next described. In chapter 2, hard-decision MIMO detection methods have been reviewed and compared in uncoded transmission systems.

A particular interest was paid to study soft-input soft-output MIMO detection methods in chapter 3. The main existing detectors have been first presented. We focused on the tree search detection methods (sphere decoder and K-Best decoder) that are able to achieve optimum performance and near optimum performance. Consequently, a LC-K-Best decoder has been proposed to avoid the full expansion and to simplify the enumeration through the use of two LUTs. We have shown that LC-K-Best decoder reduces the computational complexity of sphere decoder with an acceptable performance degradation. Then, the convergence behaviors of the iterative receivers with both coding schemes (turbo, LDPC) have been analyzed in order to retrieve the required number of inner and outer iterations. EXIT charts have been adopted for the convergence study. The simulation results have demonstrated that it is inefficient to perform a large number of inner iterations for each outer iterations. Therefore, a total number of eight iterations inside the turbo decoder distribute equally across the outer iterations have been retained. In a similar way, the LDCP decoder requires 20 iterations to converge that can be distributed in ascending order across the outer iterations. Furthermore, a comparative study in terms of performance of several MIMO detection methods and channel decoding schemes have been conducted in a Rayleigh channel model. Then several LTE multipath channel models have been considered to evaluate the performance of the proposed receiver in real propagation scenarios.

Moreover, the computational complexity of the iterative receiver in terms of number and types of float point operations has been investigated in chapter 4. In this regard,

the computational complexity of MIMO detection algorithms and channel decoder techniques (turbo, LDPC) has been evaluated and compared. In particular, we have examined the influence of the number of inner and outer iterations, and observed that there exists a trade-off to find between good performance and complexity. The simulation results demonstrate that the proposed LC-K-Best based receiver achieves the best performance-complexity trade-off among the studied receivers. We showed also that the performance and the complexity of iterative receiver with turbo decoder and LPDC decoder are highly comparable. Our study was then extended to convert our algorithm into a fixed point format. Hence, an efficient fixed point arithmetic of the proposed receiver has been presented. The simulation results demonstrate that the proposed fixed-point arithmetic does not exhibit significant performance loss with respect to the float point receiver.

In chapter 5, real-time considerations for MIMO-OFDM system have been discussed. The channel estimation of MIMO-OFDM system was first considered. The aim was to address the performance degradation of pilot-based channel estimation methods in various channel environments with different user velocities. Hence, the performance of several pilot-based channel estimation namely LS, LMMSE and TSVD were compared. LMMSE estimator improves significantly the LS estimator performance. However, it has a high computational complexity. TSVD was shown to be able to approach the performance of LMMSE using a suitable truncated threshold in a slow fading environment. However, with higher user velocities, the performance of pilot-based channel estimation is significantly degraded. Therefore, more advanced channel estimation techniques like decision directed must be considered to improve the accuracy of the channel estimates. Additionally, the synchronization of MIMO-OFDM system has been also briefly reviewed. The time synchronization in MIMO system has been performed using CAZAC sequences with cyclic delay shift between antennas. At the end, a testbed of MIMO-OFDM system using WARP platform has been used to validate the performance of the proposed approaches in real-time conditions.

Future works

In this thesis, many issues encountered by MIMO-OFDM system have been discussed. However, multiple suggestions and optimizations have not been explored and can be the topics for future study:

- Optimization of the computational complexity of the proposed iterative receiver in hardware architecture, i.g, for a high throughput and a parallel architecture more than one processing units can be investigated in order to simultaneously perform the detection and the decoding processes.
- Estimation of the power consumption of the iterative receiver is also an attractive topic specially in the increase demand for green communication systems. Additionally, latency and throughput have to be also considered in this field.
- We see that the pilot based channel estimation methods investigated in this thesis are not able to tack the variability of the channel and leads to significant performance loss. It is therefore worthy to study further advanced channel estimation techniques and to incorporate the channel estimation into the iterative process to improve the accuracy of the channel estimates.
- Design and implementation of a reconfigurable and flexible architecture which

would adaptively switch between different detectors or decoders depending on the transmission requirements and the channel environments. The architecture may involves various modulation orders and MIMO configurations. The architecture may be also designed to support multi-standards and multi-modes applications. Moreover, different levels of parallelism must be exploited in the architecture. The parallelism may be in the detector or decoder computation unit or between sub-blocks or even between frames depending on the required throughput and the complexity.

- In our study, we use a random interleaver. The random interleaver presents generally the best performance. However, it is not feasible for practical implementation. The design of an interleaver suitable for parallel and pipeline architecture without memory conflict problems is another interesting topic to be considered. The interleaver must be able to achieve near random interleaver performance and low latency.

Bibliography

[1] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," in *Proc. IEEE International Conference on Communications (ICC)*, vol. 2, pp. 1064–1070, May 1993.

[3] R. G. Gallager, *Low density parity-check codes*. PhD thesis, MIT Press, Cambridge, MA, 1963.

[4] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[5] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. IEEE International Conference on Communications (ICC) Seattle*, vol. 2, pp. 1009–1013, June 1995.

[6] J. Chen, A. Dholakia, E. Eleftheriou, M. Fossorier, and X.-Y. Hu, "Reduced-Complexity Decoding of LDPC Codes," *IEEE Transactions on Communications*, vol. 53, pp. 1288–1299, Aug. 2005.

[7] Z. Wang and G. Giannakis, "Wireless multicarrier communications," *IEEE Signal Processing Magazine*, vol. 17, pp. 29–48, May 2000.

[8] H. Zeng and L. Tong, "Blind channel estimation using the second-order statistics: asymptotic performance and limitations," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2060–2071, Aug 1997.

[9] W. V. ETTEN, "Maximum Likelihood Receiver for Multiple Channel Transmission Systems," *IEEE Transactions on Communications*, vol. 24, pp. 276–283, Feb. 1976.

[10] C. Windpassinger, *Detection and Precoding for multiple input multiple output channels*. PhD thesis, Erlangen-Nurnberg University, 2004.

[11] G. Foschini, "Layered space-time architecture for wirelless communication in fading environmnet when using multiple antennas," *Bell Labs Technical Journal*, vol. 2, pp. 41–59, Aut. 1996.

[12] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, 4 ed., 2001.

[13] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-BLAST: an architecture for realizing very high data rates overthe rich-scattering wireless channel," in *Proc. International Symposium on Signals, Systems, and Electronics*, pp. 295–300, Sept. 1998.

[14] D. Wübben, R. Böhnke, J. Rinas, V. Kühn, and K. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *Electronics Letters*, vol. 37, pp. 1348–1350, Oct. 2001.

[15] D. Wübben, R. Bohnke, V. Kuhn, and K.-D. Kammeyer, "MMSE extension of V-BLAST based on sorted QR decomposition," in *Proc. IEEE 58th Vehicular Technology Conference (VTC)*, vol. 1, pp. 508–512, Oct. 2003.

[16] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest Point Search in Lattices," *IEEE Transactions on Information Theory*, vol. 48, pp. 2201–2214, August 2002.

[17] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, vol. 44, pp. 463–471, April 1985.

[18] K.-W. Wong, C.-Y. Tsui, R. S.-K. Cheng, and W.-H. Mow, "A VLSI Architecture of a K-Best Lattice Decoding Algorithm for MIMO Channels," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 3, pp. 273–276, 2002.

[19] L. Barbero and J. Thompson, "Performance Analysis of a Fixed-Complexity Sphere Decoder in High-Dimensional Mimo Systems," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, May 2006.

[20] C. P. Schnorr and M. Euchner, "Lattice Basis Reduction: Improved Practical Algorithms and Solving Subset Sum Problems," *Math. Program*, vol. 66, pp. 181–191, 1994.

[21] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Transactions on Telecommunications*, pp. 507–511, 1995.

[22] M. Tuchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Transactions on Communications*, vol. 50, pp. 754–767, May 2002.

[23] J. W. Choi, A. Singer, J. Lee, and N. I. Cho, "Improved linear soft-input soft-output detection via soft feedback successive interference cancellation," *IEEE Transactions on Communications*, vol. 58, pp. 986–996, March 2010.

[24] L. Boher, R. Rabineau, and M. Helard, "FPGA implementationof an iterative receiver for MIMO-OFDM systems," *IEEE Journal on Selected Areas in Communications*, vol. 26, pp. 857–866, Aug. 2008.

[25] C. Studer, S. Fateh, and D. Seethaler, "ASIC Implementation of Soft-Input Soft-Output MIMO Detection Using MMSE Parallel Interference Cancellation," *IEEE Journal of Solid-State Circuits*, vol. 46, pp. 1754–1765, July 2011.

[26] I. Collings, M. Butler, and M. McKay, "Low complexity receiver design for MIMO bit-interleaved coded modulation," in *Proc. IEEE Eighth International Symposium on Spread Spectrum Techniques and Applications*, pp. 12–16, Sept. 2004.

[27] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Transactions on Communications*, vol. 51, pp. 389–399, March 2003.

[28] C. Studer and H. Bölcskei, "Soft-Input Soft-Output Single Tree-Search Sphere Decoding," *IEEE Transactions on Infromation Theory*, vol. 56, pp. 4827–4842, Oct. 2010.

[29] Z. Guo and P. Nilsson, "Algorithm and Implementation of the K-Best Sphere Decoding for MIMO Detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 491–503, Mar. 2006.

[30] R. El Chall, F. Nouvel, M. Helard, and M. Liu, "Low complexity K-Best based Iterative Receiver for MIMO systems," in *Proc. International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 451–455, Oct. 2014.

[31] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Transactions on Communications*, vol. 49, pp. 1727–1737, Oct. 2001.

[32] R. E. Chall, F. Nouvel, M. Hélard, and M. Liu, "Iterative receivers combining MIMO detection with turbo decoding: performance-complexity trade-offs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, pp. 1–19, Mar 2015.

[33] O. MK and A. H, "Channel estimation for wireless OFDM systems," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 2, pp. 18–48, 2007.

[34] J.-J. van de Beek, O. Edfors, S. M., W. S.K., and P. Ola Borjesson, "On channel estimation in OFDM systems," in *Proc. IEEE 45th Vehicular Technology Conference*, vol. 2, pp. 815–819 vol.2, Jul 1995.

[35] M. Morelli and U. Mengali, "A comparison of pilot-aided channel estimation methods for OFDM systems," *IEEE Transactions on Signal Processing*, vol. 49, pp. 3065–3073, Dec. 2001.

[36] P. Hoeher, S. Kaiser, and P. Robertson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1845–1848, Apr. 1997.

[37] Y. Zhao and A. Huang, "A novel channel estimation method for OFDM mobile communication systems based on pilot signals and transform-domain processing," in *Proc. IEEE Vehicular Technology Conference*, vol. 3, pp. 2089–2093, May 1997.

[38] S. Feng, N. Hu, B. Yang, and W. Wu, "DCT-Based Channel Estimation Method for MIMO-OFDM Systems," in *Proc. IEEE Wireless Communications and Networking Conference*, pp. 159–163, March 2007.

[39] X. Doukopoulos and R. Legouable, "Robust Channel Estimation via FFT Interpolation for Multicarrier Systems," in *Proc. IEEE Vehicular Technology Conference*, pp. 1861–1865, April 2007.

[40] M. Diallo, M. Hélard, L. Cariou, and R. Rabineau, *DFT Based Channel Estimation Methods for MIMO-OFDM Systems*, ch. 6. Vehicular Technologies: Increasing Connectivity, 2011.

[41] E. Zhou, X. Zhang, H. Zhao, and W. Wang, "Synchronization algorithms for MIMO OFDM systems," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 18–22, March 2005.

[42] H. Vikalo and B. Hassibi, "On joint detection and decoding of linear block codes on Gaussian vector channels," *IEEE Transactions on Signal Processing*, vol. 54, pp. 3330–3342, Sept. 2006.

[43] M. Al-Nuaimi and A. Siamarou, "Coherence bandwidth characterisation and estimation for indoor Rician multipath wireless channels using measurements at 62.4GHz," *IEE Proceedings Microwaves, Antennas and Propagation*, vol. 149, pp. 181–187, Jun 2002.

[44] B. Sklar, "Rayleigh fading channels in mobile digital communication systems. I. Characterization," *IEEE Communications Magazine*, vol. 35, pp. 136–146, Sep 1997.

[45] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[46] P. Elias, *Coding for noisy channels.* in IRE Convention Record, Part IV, 1955.

[47] T. J. Richardson and R. L. Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding," *IEEE Transactions on Infromation Theory*, vol. 47, pp. 599–618, Feb. 2001.

[48] LTE, *Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding.* 3GPP TS 36.212 version 11.1.0 Release 11, 2013.

[49] C. Berrou, C. Douillard, and M. Jézéquel, "Multiple parallel concatenation of circular recursive convolutional (CRSC) codes," *Annals of Telecommun*, vol. 54, no. 3-4, pp. 166–172, 1989.

[50] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE Global Telecommunications Conference*, pp. 1680–1686, Nov. 1989.

[51] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Infromation Theory*, vol. 42, pp. 429–445, Mar. 1996.

[52] A. Matache, S. Dolinar, and F. Pollara, "Stopping Rules for Turbo Decoders," tech. rep., TMO Progress Report 42-142, August 2000.

[53] C. Schurgers, F. Catthoor, and M. Engels, "Memory optimization of MAP turbo decoder algorithms," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, pp. 305–312, April 2001.

[54] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, pp. 533–547a, September 1981.

[55] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 638–656, 2001.

[56] A. Anastasopoulos, "A comparison between the sum-product and the min-sum iterative detection algorithms based on density evolution," in *Proc. IEEE Global Telecommunications Conference*, vol. 2, pp. 1021–1025, 2001.

[57] L. Zheng and D. N. C. Tse, "Diversity and Multiplexing : a Fundamental Trade-off in Multiple-Antenna Channels," *IEEE Transactions on Information Theory*, vol. 49, pp. 1073–1096, May 2003.

[58] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication : performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744–765, Mars 1998.

[59] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, Oct. 1998.

[60] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-Time Block Codes from Orthogonal Designs," *IEEE Transactions on Information Theory*, vol. 45, pp. 744–765, July 1999.

[61] O. Tirkkonen, A. Boariu, and A. Hottinen, "Minimal non-orthogonality rate 1 space-time block code for 3+ Tx antennas," in *Proc. IEEE Sixt International Symposium on Spread Spectrum Techniques and Applications*, vol. 2, pp. 429–432, 2000.

[62] H. Jafarkhani, "A quasi-orthogonal space-time block code," *IEEE Transactions on Communications*, vol. 49, pp. 1–4, Jan 2001.

[63] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, pp. 927–946, May 1998.

[64] B. Hassibi and B. M. Hochwald, "High-rate codes that are linear in space and time," *IEEE Transactions on Information Theory*, vol. 48, pp. 1804–1824, July 2002.

[65] E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, vol. 10, pp. 585–595, December 1999.

[66] B. Hochwald and W. Sweldens, "Differential unitary space-time modulation," *IEEE Transactions on Communications*, vol. 48, no. 12, pp. 2041–2052, 2000.

[67] H. Jafarkhani and V. Tarokh, "Multiple transmit antenna differential detection from generalized orthogonal designs," *IEEE Transactions on Information Theory*, vol. 47, no. 6, pp. 2626–2631, 2001.

[68] J. Boutros, F. Boixadera, and C. Lamy, "Bit-interleaved coded modulations for multiple-input multiple-output channels," in *Proc. IEEE Sixth International Symposium on Spread Spectrum Techniques and Applications*, vol. 1, pp. 123–126, Sept. 2000.

[69] M. Myllyla, M. Juntti, and J. R. Cavallaro, "Implementation aspects of list sphere decoder algorithms for MIMO-OFDM systems," *Signal Processing*, vol. 90, no. 10, pp. 2863–2876, 2010.

[70] L. Lovasz, *An Algorithmic Theory of Numbers, Graphs and Convexity.* Society for Indestrial and Applied Mathematics, 1987.

[71] A. Burg, N. Felber, and W. Fichtner, "A 50 mbps 4×4 maximum likelihood decoder for multiple-input multiple-output systems with QPSK modulation," in *Proc. IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, vol. 1, pp. 322–335, Dec. 2003.

[72] D. Garrett, L. Davis, S. ten Brink, and B. Hochwald, "APP processing for high performance MIMO systems receiver symbol detector," in *Proc. IEEE Custom Integrated Circuits Conference*, pp. 271–274, Sept 2003.

[73] B. Hassibi, "An efficient square-root algorithm for BLAST," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 737–740, 2000.

[74] H. Zhu, Z. Lei, and F. Chin, "An improved square-root algorithm for BLAST," *IEEE Signal Processing Letters*, vol. 11, pp. 772–775, 2004.

[75] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*. Siam, 2002.

[76] S. Aubert, M. Mohaisen, F. Nouvel, and K. Chang, "Parallel QR Decomposition in LTE-A Systems," in *Proc. IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, May 2010.

[77] G. J. Foschini, G. D. Golden, A. Valenzela, and P. W. Wolniansky., "Simplified Processing for High Spectral Efficiency Wireless Communications Emplying Multi-Element Arrays," *IEEE Journal on Selectes Areas in Commununications*, vol. 17, pp. 1841–1852, November 1999.

[78] E. Viterbo and J. Boutros, "A Universal Lattice Code Decoder for Fading Channels," *IEEE Transactions on Information Theory*, vol. 45, pp. 1639–1642, July 1999.

[79] M. Damen, H. E. Gamal, and G. Caire, "On Maximum-Likelihood Detection and the Search for the Closest Lattice Point," *IEEE Transactions on Information Theory*, vol. 49, pp. 2389–2402, Oct. 2003.

[80] B. Hassibi and H. Vikalo, "On the expected complexity of sphere decoding," in *Proc. Thirty-Fifth Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1051–1055, Nov. 2001.

[81] M. Pohst, "On the computation of lattice vectors of minimal length, successive minima and reduced bases with applications," *ACM SIGSAM Bulletin*, vol. 15, pp. 37–44, Feb 1981.

[82] W. H. Mow, *Maximum likelihood sequence estimation from the lattice viewpoint*. PhD thesis, Chinese University of Hong Kong, Department of Information Engineering, 1991.

[83] W. H. Mow, "Maximum likelihood sequence estimation from the lattice viewpoint," *IEEE Transactions on Information Theory*, vol. 40, pp. 1591–1600, Sep 1994.

[84] E. Viterbo and E. Biglieri, "A universal decoding algorithm for lattice codes," *Colloque GRETSI*, vol. 14, pp. 611–614, Sept. 1993.

[85] Z. Guo and P. Nilsson, "Reduced complexity Schnorr-Euchner decoding algorithms for MIMO systems," *IEEE Communications Letters*, vol. 8, pp. 286–288, May 2004.

[86] B. Hassibi and H. Vikalo, "On the Sphere-Decoding Algorithm I. Expected Complexity," *IEEE Transactions on Signal Processing*, vol. 53, pp. 2806–2818, Aug. 2005.

[87] J. Zhang and M. Fossorier, "Shuffled Turbo Decoding," *IEEE Transactions on Communications*, vol. 53, pp. 209–213, Feb. 2005.

[88] R. Gowaikar and B. Hassibi, "Statistical Pruning for Near-Maximum Likelihood Decoding," *IEEE Transactions on Signal Processing*, vol. 55, pp. 2661–2675, June 2007.

[89] B. Shim and I. Kang, "Sphere Decoding With a Probabilistic Tree Pruning," *IEEE Transactions on Signal Processing*, vol. 56, pp. 4867–4878, Oct. 2008.

[90] J. Ahn, H.-N. Lee, and K. Kim, "Schnorr-euchner sphere decoder with statistical pruning for MIMO systems," in *Proc. 6th International conference on Symposium on Wireless Communication Systems*, pp. 619–623, 2009.

[91] J. Jaldén and B. Ottersten, "Parallel Implementation of a Soft Output Sphere Decoder," in *Proc. Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, pp. 581–585, Nov. 2005.

[92] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE Journal on Solid-State Circuits*, vol. 40, pp. 1566–1577, July 2005.

[93] L. Barbero and J. Thompson, "FPGA Design Considerations in the Implementation of a Fixed-Throughput Sphere Decoder for MIMO Systems," in *Proc. International Conference on Field Programmable Logic and Applications*, Aug 2006.

[94] J. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Transactions on communications*, vol. 32, pp. 169–176, Feb 1984.

[95] S. Baro, J. Hagenauer, and M. Witzke, "Iterative detection of MIMO transmission using a list-sequential (LISS) detector," in *Proc. IEEE International Conference on Communications*, vol. 4, pp. 2653–2657, May 2003.

[96] J. Hagenauer and C. Kuhn, "The List-Sequential (LISS) Algorithm and Its Application," *IEEE Transactions on Communications*, vol. 55, pp. 918–928, May 2007.

[97] S. Chen, T. Zhang, and Y. Xin, "Relaxed K-Best MIMO Signal Detector Design and VLSI Implementation," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, pp. 328–337, Mar. 2007.

[98] M. Shabany and P. Gulak, "Scalable VLSI architecture for K-best lattice decoders," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 940–943, May 2008.

[99] M. Mahdavi and M. Shabany, "Novel MIMO Detection Algorithm for High-Order Constellations in the Complex Domain," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, pp. 834–847, May 2013.

[100] A. Wiesel, X. Mestre, A. Pages, and J. Fonollosa, "Efficient implementation of sphere demodulation," in *Proc. 4th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 36–40, Jun. 2003.

[101] Y. L. C. de Jong and T. J. Willink, "Iterative Tree Search Detection for MIMO Wireless Systems," *IEEE Transactions on Communications*, vol. 53, pp. 930–935, June 2005.

[102] M. Wenk, M. Zellweger, A. Burg, N. Felber, and W. Fichtner, "K-Best MIMO detection VLSI architectures achieving up to 424 Mbps," in *Proc. IEEE International Symposium on Circuits and Systems(ISCAS)*, p. 1154, May 2006.

[103] M. Shabany and P. Gulak, "A 675 Mbps, 4×4 64-QAM K-Best MIMO Detector in 0.13μm CMOS," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, pp. 135–147, Jan 2012.

[104] Q. Li and Z. Wang, "Reduced Complexity K-Best Sphere Decoder Design for MIMO Systems," *Circuits Systems and Signal Processing*, vol. 27, pp. 491–505, August 2008.

[105] M. Mohaisen, K. Chang, and B. Koo, "Adaptive Parallel and Iterative QRDM Algorithms for Spatial Multiplexing MIMO Systems," in *Proc. IEEE 70th Vehicular Technology Conference Fall*, pp. 1–5, Sept 2009.

[106] A. Burg, M. Borgmanr, M. Wenk, C. Studer, and H. Bolcskei, "Advanced receiver algorithms for MIMO wireless communications," in *Proc. Design, Automation and Test in Europe*, vol. 1, pp. 6 pp.–, March 2006.

[107] M. Shabany and P. G. Gulak, "A 0.13um CMOS, 655Mb/s 4»4 64-QAM K-Best MIMO detector," in *Proc. IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 256–257,257a, Feb. 2009.

[108] L. Liu, F. Ye, X. Ma, T. Zhang, and J. Ren, "A 1.1-Gb/s 115-pJ/bit Configurable MIMO Detector using 0.13um CMOS Technology," *IEEE Transactions on Circuits and Systems II :Express Briefs*, vol. 57, pp. 701–705, Sept 2010.

[109] L. Barbero and J. Thompson, "Rapid Prototyping of a Fixed-Throughput Sphere Decoder for MIMO Systems," in *Proc. IEEE International Conference on Communications (ICC)*, vol. 7, pp. 3082–3087, June 2006.

[110] L. Barbero and J. Thompson, "A Fixed-Complexity MIMO Detector Based on the Complex Sphere Decoder," in *Proc. IEEE 7th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, July 2006.

[111] J. Jaldén, L. G. Barbero, B. Ottersten, and J. S. Thompson, "Full diversity detection in MIMO systems with a fixed complexity sphere decoder," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, pp. 49–52, Avril 2006.

[112] M. Mohaisen and K. Chang, "On Improving the Efficiency of the Fixed-Complexity Sphere Decoder," in *Proc. IEEE 70th Vehicular Technology Conference Fall*, pp. 1–5, Sept 2009.

[113] C. Xiong, X. Zhang, K. Wu, and D. Yang, "A simplified fixed-complexity sphere decoder for V-BLAST systems," *IEEE Communications Letters*, vol. 13, pp. 582–584, August 2009.

[114] M. Khairy, M. Abdallah, and S.-D. Habib, "Efficient FPGA Implementation of MIMO Decoder for Mobile WiMAX System," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–5, June 2009.

[115] H. Yao and G. W. Wornell, "Lattice-Reduction-Aided Detectors for MIMO Communication Systems," in *Proc. IEEE Global Telecommunications Conference*, vol. 1, pp. 424–428, Nov. 2002.

[116] C. Windpassinger and R. F. H. Fischer, "Low-Complexity Near-Maximum-Likelihood Detection and precodind for MIMO system using lattice reduction," in *Proc. IEEE Information Theory Workshop*, pp. 345–348, March 2003.

[117] D. Wübben, R. Bohnke, V. Kuhn, and K.-D. Kammeyer, "MMSE-based lattice-reduction for near-ML detection of MIMO systems," in *Proc. ITG Workshop on Smart Antennas*, pp. 106–113, March 2004.

[118] B. Lamacchia, *Basis Reduction Algorithms and Subset Sum Problems*. PhD thesis, Massachusetts Institute of Technology, 1991.

[119] A. K. Lenstra, H. W. Lenstra, Jr., , and L. Lovasz, "Factoring Polynomials with Rational Coefficients," *Mathematische Annalen*, vol. 261, pp. 515–534, 1982.

[120] M. Seysen, "Simultaneous reduction of a lattice basis and its reciprocal basis," *Combinatorica*, vol. 13, no. 3, pp. 363–376, 1993.

[121] Y. H. Gan, C. Ling, and W. H. Mow, "Complex Lattice Reduction Algorithm for Low-Complexity Full-Diversity MIMO Detection," *IEEE Transactions on Signal Processing*, vol. 57, pp. 2701–2710, July 2009.

[122] D. Wübben and D. Seethaler, "On the Performance of Lattice Reduction Schemes for MIMO Data Detection," in *Proc. Forty-First Asilomar Conference on Signals, Systems and Computers*, pp. 1534–1538, Nov 2007.

[123] L. Barbero, T. Ratnarajah, and C. Cowan, "A comparison of complex lattice reduction algorithms for MIMO detection," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2705–2708, March 2008.

[124] S. Aubert and M. Mohaisen, *From Linear Equalization to Lattice-Reduction-Aided Sphere-Detector as an Answer to the MIMO Detection Problematic in Spatial Multiplexing Systems*, ch. 5, p. 448. Vehicular Technologies: Increasing Connectivity, Feb 2011.

[125] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless communications*. Cambridge Univ. Press, 2003.

[126] A. Glavieux, C. Laot, and J. Labat, "Turbo Equalization over a frequency selective channel," in *Proc. 1st Symp. Turbo Codes*, pp. 96–102, 1997.

[127] M. Tuchler, A. C. Singer, and R. Kotter, "Minimum mean squared error equalization using *a priori* information," *IEEE Transactions on Signal Processing*, vol. 50, pp. 673–683, Mar. 2002.

[128] C. Laot, R. L. Bidan, and D. Leroux, "Low-Complexity MMSE Turbo Equalization: A Possible Solution for EDGE," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 965–974, May 2005.

[129] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 47, no. 7, pp. 1046–1061, Jul.1999.

[130] M. Witzke, S. Baro, F. Schreckenbach, and J. Hagenauer, "Iterative detection of MIMO signals with linear detectors," in *Proc. Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 289–293, Nov 2002.

[131] P.-J. Bouvet and M. Hélard, "Optimal space-time coding under iterative processing," *annals of telecommunications*, vol. 69, pp. 229–238, 2014.

[132] H. Lee, B. Lee, and I. Lee, "Iterative detection and decoding with an improved V-BLAST for MIMO-OFDM systems," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 504–513, March 2006.

[133] C. Studer, A. Burg, and H. Bölcskei, "Soft-Output Sphere Decoding: Algorithms and VLSI Implementation," *IEEE Journal on Selected Areas in Communications*, vol. 26, pp. 290–300, Feb. 2008.

[134] E. P. Adeva, T. Seifert, and G. Fettweis, "VLSI Architecture for MIMO Soft-Input Soft-Output Sphere Detection," *Journal of Signal Processing Systems*, vol. 70, pp. 125–143, Feb. 2013.

[135] L. Barbero, T. Ratnarajah, and C. Cowan, "A low-complexity soft-MIMO detector based on the fixed-complexity sphere decoder," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2669–2672, Apr. 2008.

[136] B. Wu and G. Masera, "Efficient VLSI implementation of soft-input soft-output fixed-complexity sphere decoder," *IET Communications*, vol. 6, pp. 1111–1118, June 2012.

[137] E. M. Witte, F. Borlenghi, G. Ascheid, R. Leupers, and H. Meyr, "A Scalable VLSI Architecture for Soft-Input Soft-Output Single Tree-Search Sphere Decoding," *IEEE Transactions on Circuits and Systems II*, vol. 57, pp. 706–710, Sept. 2010.

[138] F. Borlenghi, E. Witte, G. Ascheid, H. Meyr, and A. Burg, "A 772Mbit/s 8.81bit/nJ 90nm CMOS soft-input soft-output sphere decoder," in *Proc. IEEE Asian Solid State Circuits Conference (A-SSCC)*, pp. 297–300, Nov. 2011.

[139] Y. Sun and J. Cavallaro, "Trellis-Search Based Soft-Input Soft-Output MIMO Detector: Algorithm and VLSI Architecture," *IEEE Transactions on Signal Processing*, vol. 60, pp. 2617–2627, May 2012.

[140] H.-C. Wang, T.-L. Liu, Y.-W. Wu, , and H.-P. Ma, "A Power-Efficient Soft-Output Detector for Spatial-Multiplexing MIMO Communications," *Journal of Electrical and Computer Engineering*, vol. 2012, p. 9, 2012.

[141] L. Liu, "High-throughput hardware-efficient soft-input soft-output MIMO detector for iterative receivers," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2151–2154, May 2013.

[142] X. Chen, G. He, and J. Ma, "VLSI Implementation of a High-Throughput Iterative Fixed-Complexity Sphere Decoder," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 60, pp. 272–276, May 2013.

[143] D. Patel, V. Smolyakov, M. Shabany, and P. G. Gulak, "VLSI implementation of a WiMAX/LTE compliant low-complexity high-throughput soft-output K-Best MIMO detector," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 593–596, June 2010.

[144] P.-Y. Tsai, W.-T. Chen, X.-C. Lin, and M.-Y. Huang, "A 4»4 64-QAM Reduced-Complexity K-Best MIMO Detector up to 1.5Gbps," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 3953–3956, May 2010.

[145] D. Seethaler, G. Matz, and F. Hlawatsch, "An efficient MMSE- based demodulator for MIMO bit-interleaved coded modulation," in *Proc. IEEE Global Telecommunications Conference*, vol. 4, pp. 2455–2459, Nov. 2004.

[146] C. Studer, *Iterative MIMO Decoding: Algorithms and VLSI Implementation Aspects*. PhD thesis, ETH Zurich, Switzerland, Series in Microelectronics, 2009.

[147] E. Zimmermann and G. Fettweis, "Adaptive vs. Hybrid Iterative MIMO Receivers Based on MMSE Linear and Soft-SIC Detection," in *Proc. International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–5, Sept. 2006.

[148] E. Zimmermann, *Complexity Aspects in Near-Capacity MIMO Detection-Decoding*. PhD thesis, Technische Universitat Dresden, 2007.

[149] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoder," *IEEE Transactions on Wireless Communication*, vol. 3, pp. 2299–2311, Nov. 2004.

[150] E. Zimmermann and G. Fettweis, "Unbiased MMSE Tree Search Detection for Multiple Antenna Systems," in *Proc. International Symposium on Wireless Personel Mutimedia Communications*, Sept. 2006.

[151] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, "Soft-input soft-output lattice sphere decoder for linear channels," in *IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 3, pp. 1583–1587, Dec. 2003.

[152] M. Wenk, A. Burg, M. Zellwege, C. Studer, and W. Fichtner, "VLSI implementation of the list sphere algorithm," in *Proc. 24th Norchip Conference*, pp. 107–110, Nov. 2006.

[153] M. Samuel and M. Fitz, "Iterative Sphere Detectors Based on the Schnorr-Euchner Enumeration," *IEEE Transactions on Wireless Communications*, vol. 9, pp. 2137–2144, July 2010.

[154] C.-H. Liao, T.-P.Wang, and T. D. Chiueh, "A 74.8 mW soft-output detector IC for 8×8 spatial-multiplexing MIMO communications," *IEEE Journal on Solid-State Circuits*, vol. 45, pp. 411–421, Feb. 2010.

[155] M. Wenk, L. Bruderer, A. Burg, and C. Studer, "Area- and throughput-optimized VLSI architecture of sphere decoding," in *Proc. 18th IEEE/IFIP VLSI System on Chip Conference (VLSI-SoC)*, pp. 189–194, Sept 2010.

[156] B. Mennenga, A. von Borany, and G. Fettweis, "Complexity Reduced Soft-In Soft-Out Sphere Detection Based on Search Tuples," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–6, June 2009.

[157] D. Milliner, E. Zimmermann, J. Barry, and G. Fettweis, "Channel state information based LLR clipping in list MIMO detection," in *Proc. IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, Sept 2008.

[158] E. Zimmermann, D. Milliner, J. Barry, and G. Fettweis, "Optimal LLR Clipping Levels for Mixed Hard/Soft Output Detection," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, pp. 1–5, Nov 2008.

[159] S. Aubert, A. Anchora, and F. Nouvel, "Multi-Level Log-Likelihood Ratio Clipping in a Soft-Decision Near-Maximum Likelihood Detector," in *Proc. International conference on digital telecommunications*, pp. 30–35, April 2011.

[160] R. Wang and G. B. Giannakis, "Approaching MIMO Channel Capacity with Reduced-Complexity Soft Sphere Decoding," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 3, pp. 1620–1625, Mar. 2004.

[161] O. Paker, S. Eckert, and A. Bury, "A low cost multi-standard near-optimal soft-output sphere decoder: Algorithm and architecture," in *Proc. Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1402–1407, March 2010.

[162] I. J. Choi, B. Shim, J. Nelson, and A. Singer, "Efficient soft-input soft-output MIMO detection via improved M-algorithm," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–5, May 2010.

[163] B. Widdup, G. Woodward, and G. Knagge, "A highly-parallel VLSI architecture for a list sphere detectors," in *Proc. IEEE International Conference on Communications*, vol. 5, pp. 2720–2725, June 2004.

[164] S. Bittner, E. Zimmermann, W. Rave, and G. Fettweis, "List Sequential MIMO Detection: Noise Bias Term and Partial Path Augmentation," in *Proc. IEEE International Conference on Communications*, vol. 3, pp. 1300–1305, June 2006.

[165] P. Luethi, A. Burg, S. Haene, D. Perels, N. Felber, and W. Fichtner, "VLSI Implementation of a High-Speed Iterative Sorted MMSE QR Decomposition," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1421–1424, May 2007.

[166] M. Mayer, M. Simko, and M. Rupp, "Soft-Output Sphere Decoding : Single tree search vs. improved K-best," in *Proc. 18th International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2011.

[167] D. Milliner, E. Zimmermann, J. Barry, and G. Fettweis, "A Fixed-Complexity Smart Candidate Adding Algorithm for Soft-Output MIMO Detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, pp. 1016–1025, Dec. 2009.

[168] L. G. Barbero and J. S. Thompson, "Extending a Fixed-Complexity Sphere Decoder to Obtain Likelihood Information for Turbo-MIMO Systems," *IEEE Transactions on Vehicular Technology*, vol. 57, pp. 2804–2814, Sept. 2008.

[169] D. L. Milliner and J. R. Barry, "A Lattice-Reduction-Aided Soft Detector for Multiple-Input Multiple-Output Channels," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, pp. 1–5, Nov 2006.

[170] V. Ponnampalam, D. McNamara, A. Lillie, and M. Sandell, "On Generating Soft Outputs for Lattice-Reduction-Aided MIMO Detection," in *Proc. IEEE International Conference on Communications*, pp. 4144–4149, June 2007.

[171] W. Zhang and X. Ma, "Approaching Optimal Performance By Lattice-Reduction Aided Soft Detectors," in *Proc. 41st Annual Conference on Information Sciences and Systems*, pp. 818–822, March 2007.

[172] C.-H. Liao, I.-W. Lai, K. Nikitopoulos, F. Borlenghi, D. Kammler, M. Witte, D. Zhang, T.-D. Chiueh, G. Ascheid, and H. Meyr, "Combining orthogonalized partial metrics: Efficient enumeration for soft-input sphere decoder," in *Proc. IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1287–1291, Sept. 2009.

[173] S. ten Brink, "Convergence of iterative decoding," *Electronics Letters*, vol. 35, pp. 806–808, May 1999.

[174] J. Hagenauer, "The exit chart - introduction to extrinsic information transfer in iterative processing," in *Proc 12th European Signal Processing Conference*, pp. 1541–1548, Sept 2004.

[175] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Transactions on Communications*, vol. 52, pp. 670–678, April 2004.

[176] E. Sharon, A. Ashikhmin, and S. Litsyn, "Analysis of low-density parity-check codes based on EXIT functions," *IEEE Transactions on Communications*, vol. 54, pp. 1407–1414, Aug 2006.

[177] E. Sharon, S. Litsyn, and J. Goldberger, "An efficient message-passing schedule for LDPC decoding," in *Proc. IEEE Convention of Electrical and Electronics Engineers*, pp. 223–226, Sept. 2004.

[178] P.-J. Bouvet, *Récepteurs itératifs pout systèmes multi-antennes*. PhD thesis, INSA Rennes, 2005.

[179] J.-C. Guey, M. Fitz, M. Bell, and W.-Y. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 47, pp. 527–537, Apr 1999.

[180] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Transactions on Broadcasting*, vol. 48, pp. 223–229, Sep 2002.

[181] M. Henkel, C. Schilling, and W. Schroer, "Comparison of Channel Estimation Methods for Pilot Aided OFDM Systems," in *Proc. IEEE 65th Vehicular Technology Conference*, pp. 1435–1439, April 2007.

[182] L. Yu and S. Yang, "Blind Channel Estimation of MIMO-OFDM by Virtual Carriers," in *Proc. International Workshop on Signal Design and Its Applications in Communications*, pp. 299–302, Sept 2007.

[183] C. Shin, R. Heath, and E. Powers, "Blind Channel Estimation for MIMO-OFDM Systems," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 670–685, March 2007.

[184] Y. Zeng and T.-S. Ng, "A semi-blind channel estimation method for multiuser multiantenna OFDM systems," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1419–1429, May 2004.

[185] B. Ozbek, D. Le Ruyet, and C. Panazio, "Pilot-symbol-aided iterative channel estimation for OFDM-based systems," in *Proc. 13th European Signal Processing Conference*, pp. 1–4, Sept 2005.

[186] Y. Li, N. Seshadri, and S. Ariyavisitakul, "Channel estimation for OFDM systems with transmitter diversity in mobile wireless channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 461–471, Mar 1999.

[187] I. Barhumi, G. Leus, and M. Moonen, "Optimal training design for MIMO OFDM systems in mobile wireless channels," *IEEE Transactions on Signal Processing*, vol. 51, pp. 1615–1624, June 2003.

[188] LTE, *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation.* 3GPP TS 36.211 version 11.1.0 Release 11, 2013.

[189] P. Hoeher, S. Kaiser, and P. Robertson, "Pilot-Symbol-Aided Channel Estimation in Time and Frequency," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, pp. 90–96, 1997.

[190] E. O, S. M, van de Beek JJ, W. S, and B. P, "OFDM channel estimation by singular value decomposition," *IEEE Transactions on communications*, vol. 46, no. 7, pp. 931–939, 1998.

[191] S. Sun, I. Wiemer, C. Ho, and T. Tjhung, "Training sequence assisted channel estimation for MIMO-OFDM," in *Proc. IEEE Wireless Communications and Networking*, vol. 1, pp. 38–43, March 2003.

[192] M. Diallo, R. Rabineau, and L. Cariou, "Robust DCT Based Channel Estimation for MIMO-OFDM System," in *Proc. IEEE Wireless Communications and Networking Conference*, pp. 1–5, April 2009.

[193] M. Diallo and M. Hèlard, "Channel Estimation methods with low complexity for 3GPP/LTE," *ARIMA Journal*, vol. 18, pp. 93–116, 2014.

[194] B. L. Saux, *Estimation de canal pour Systèmes multi-antennes et multi-porteuses.* PhD thesis, INSA Rennes, 2007.

[195] M. Diallo, *Etude et mise en oeoeuvre module d'estimation de canal pour les systèmes MIMO-OFDM.* PhD thesis, INSA Rennes, 2010.

[196] J. Romano and J. Lopez, "Practical application of transfer path analysis to resolve structure-borne noise problems in vehicle design," in *Proc. 21st International Seminar on Modal Analysis*, 1996.

[197] G. Tauböck and F. Hlawatsch, "A compressed sensing technique for OFDM channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2885–2888, March 2008.

[198] C. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of compressive sensing to sparse channel estimation," *IEEE Communications Magazine*, vol. 48, pp. 164–174, November 2010.

[199] X. Deng, A. Haimovich, and J. Garcia-Frias, "Decision directed iterative channel estimation for MIMO systems," in *Proc. IEEE International Conference on Communications*, vol. 4, pp. 2326–2329 vol.4, May 2003.

[200] M. Loncar, R. Müller, J. Wehinger, C. Mecklenbraüker, and T. Abe, "Iterative channel estimation and data detection in frequencyselective fading MIMO channels," *European Transactions on Telecommunications*, pp. 459–470, 2004.

[201] B. L. Saux and M. Hèlard, "Iterative Channel Estimation for 2x2 and 4x2 non orthogonal MIMO schemes," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1253–1257, March 2007.

[202] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[203] Y. Xie and C. Georghiades, "Two EM-type channel estimation algorithms for OFDM with transmitter diversity," *IEEE Transactions on Communications*, vol. 51, pp. 106–115, Jan 2003.

[204] J. Fessler and A. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Transactions on Signal Processing*, vol. 42, pp. 2664–2677, Oct 1994.

[205] J. Ylioinas and M. Juntti, "Iterative Joint Detection, Decoding, and Channel Estimation in Turbo-Coded MIMO-OFDM," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 1784–1796, May 2009.

[206] C. Komninakis, C. Fragouli, A. Sayed, and R. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Transactions on Signal Processing*, vol. 50, pp. 1065–1076, May 2002.

[207] J. Choi, M. Bouchard, and T. H. Yeap, "Adaptive filtering-based iterative channel estimation for MIMO wireless communications," in *Proc. IEEE International Symposium on Circuits and Systems*, pp. 4951–4954, May 2005.

[208] http://warpproject.org/trac/wiki/HardwareUsersGuides/WARPv3.

[209] A. Mody and G. Stuber, "Synchronization for MIMO OFDM systems," in *Proc. IEEE Global Telecommunications Conference*, vol. 1, pp. 509–513, 2001.

[210] T.-J. Liang, X. Li, R. Irmer, and G. Fettweis, "Synchronization in OFDM-based WLAN with transmit and receive diversities," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, pp. 740–744, Sept 2005.

[211] C. Feng, J. Zhang, Y. Zhang, and M. Xia, "A Novel Timing Synchronization Method for MIMO OFDM Systems," in *Proc. IEEE Vehicular Technology Conference*, pp. 913–917, May 2008.

[212] D. Chu, "Polyphase codes with good periodic correlation properties," *IEEE Transactions on Information Theory*, vol. 18, pp. 531–532, Jul 1972.

[213] http://www.xilinx.com/support/documentation/virtex-6.htm.

INSA de RENNES
Service des Formations

AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

Titre de la thèse:
Récepteur itératif pour les systèmes MIMO-OFDM basé sur le décodage sphérique: Convergence, performance et complexité

Nom Prénom de l'auteur : EL CHALL RIDA

Membres du jury :
- Madame HELARD Maryline
- Monsieur BERDER Olivier
- Madame DOUILLARD Catherine
- Madame NOUVEL Fabienne
- Monsieur JEGO Christophe
- Monsieur GELLE Guillaume

Président du jury : Guillaume GELLE

Date de la soutenance : 22 Octobre 2015

Reproduction de la these soutenue

Thèse pouvant être reproduite en l'état
~~Thèse pouvant être reproduite après corrections suggérées~~

Fait à Rennes, le 22 Octobre 2015

Signature du président de jury

Le Directeur,

M'hamed DRISSI